Vprašanja so bila pobrana z elektrina.net, odgovori pa so moje delo. Odgovore sem dobil v skriptah, knjigah, nekaj idej pa sem si sposodil s foruma. Odgovori niso neka znanstvena razlaga, ampak v večini primerov kar »kmečka«. Za odgovore ne garantiram, lahko da sem kje tudi popolnoma zabluziu!!!

1. ***Izpelji obrazec za pretvorbo iz PKNO v PDNO + kaj je negacija maksterma***

$$\overline{M_i} = m_{2^n-1-i}$$

Negacija maksterma z index-om i je minterm z index-om $2^n - 1 - i$.

2. ***napisi enacbe za mealeyev in moorov avtomat, kaksne so razlike,koliko razlicnih izhodov lahko pri mealyjevem avtomatu dobis v enem urinem ciklu ce ga prozimo na negativno fronto***

mealey:

$A = \{X, S, Z, \delta, \lambda\}$

$\delta : X \times S \rightarrow S$

$\lambda : X \times S \rightarrow Z$

$\Delta^1 s = \delta(s, x)$

$z = \lambda(s, x)$

moore:

$A = \{X, S, Z, \delta, \lambda\}$

$\delta : X \times S \rightarrow S$

$\lambda : X \rightarrow Z$

$\Delta^1 s = \delta(s, x)$

$z = \lambda(s)$

Izhodi Mealeyevega avtomata so odvisni od stanja in vhodnih spremenljivk, z razliko od Moorovega avtomata, kri katerem so izhodi odvisni samo od stanj. Izhodi Mealeyevega avtomata prehitevajo Moorovega za pol takta- izhodi Mealeya se spremenijo sočasno s prehodom v stanje, medtem ko se pri Moorovem spremenijo šele ko je avtomat že v stanju.

3. ***izpelji relacijo med shefferjem in piercem pa se neke poenostavitve, tko da dobis [cenzura] realizirano na enem oz. dveh nivojih***

$$\overline{x_1 \downarrow x_2} = \overline{x_1} \mid \overline{x_2} \Leftrightarrow \overline{x_1 \mid x_2} = \overline{x_1} \downarrow \overline{x_2}$$

(negacija pierca = sheffer negiranih spremenljivk in obratno)

4. ***Izolirano stanje stevca(kaj je to, kdaj lahko nastopi)***

Izolirano stanje števca je stanje, ki ni v delovnem ciklu. Števec tako realiziramo, da če se števec znajde v tem stanju, preide v začetno stanje (???)

5. Izpelji enačbe za prenose pri paralelnem seštevalniku s PG modulom (neki tazga)

$P_i = a_i \oplus b_i$
$G_i = a_i b_i$

$C_{i-1} = C_{vh}$
$C_0 = G_o + C_{vh} P_0$
$C_1 = G_1 + C_0 P_1$
$C_2 = G_2 + C_1 P_2$
…in tako naprej
$C_k = G_k + C_{k-1} P_k$

6. Izpelji enacbo za T celico z g parametri in jo realiziraj z vrati

Splošna pomnilna enačba:
$Q(t+1) = g_1 Q(t) + g_2 \overline{Q(t)}$

T – triggering – proženje
Celica pri logični 0 ohranja, pri logični 1 pa spremeni stanje.

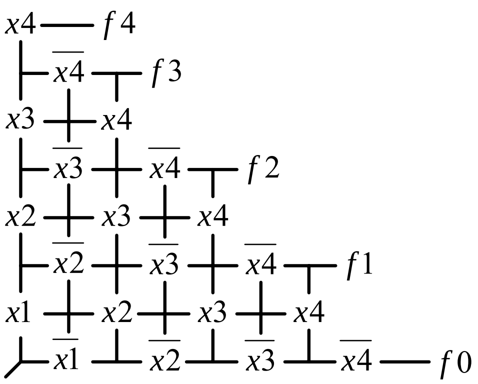
S pomočjo splošne pomnilne enačbe in opisa, kaj T celica dela, sestavimo vzbujalno tabelo:

g1	g2	Q(t)	Q(t+1)	T
0	0	0	0	0
0	0	1	0	1
0	1	0	1	1
0	1	1	0	1
1	0	0	0	0
1	0	1	1	0
1	1	0	1	1
1	1	1	1	0

Narišemo veitchev diagram za T, minimiziramo in dobimo enačbo za $T = \overline{g_1} Q(t) + g_2 \overline{Q(t)}$.

7. realizacija simetrienih funkcij s Caldwellovim vezjem in bistabilni dvojeek

Realizacija simetričnih funkcij z navadnimi operatorji v dvonivojski izvedbi je zelo obsežna, zato uporabljamo rešitev v relejski tehniki – Caldwellovo preklopno vezje (primer):



8. prepovedana stanja v pomnilniških celicah

Pri nesinhroniziranih celicah je prepovedano stanje (ker je $Q=\overline{Q}$), pri sinhroniziranih pa nedoločeno stanje (lahko je izhod 0 ali 1).
RS ff realizirana z NOR: 1,1.
RS ff realizirana z NAND: 0,0.

9. razlike med JK in RS (treba je bilo upoštevati posamezne zakasnitve, nestabilna stanja, itd.).

JK in SR celici sta si zelo podobni, obe se pri kombinacijah 00, 01, 10 enako odzivajo. Razlika je pri kombinaciji 11, pri kateri ima RS celica prepovedano stanje, zato ker bi se v tem primeru na izhodih pojavilo $Q=\overline{Q}$, kar pa nikakor ne more biti res. Celica JK pa pri tej kombinaciji deluje normalno. (po mojem mnenju ima JK celica nekoliko daljši zakasnitveni čas, zato lahko pravilno deluje pri nižjem taktu od RS)

10. ortogonalna funkcija – definicija

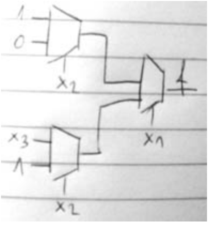
Funkcija je ortogonalna, če velja $k_i k_j = 0$ za vsak par iz disjunktivnega zapisa funkcije. Vsaka PDNO funkcija je ortogonalna. Minimizirana disjunktivna funkcija pa je ortogonalna samo, če je bila vsaka enica v Veitchu uporabljena samo enkrat.

11. realizacija popolnega seštevalnika z NAND

Enačbe za popolni seštevalnik:
$S_i = A_i B_i C_{i-1} + A_i \overline{B_i}\,\overline{C_{i-1}} + \overline{A_i}\,\overline{B_i} C_{i-1} + \overline{A_i} B_i \overline{C_{i-1}}$
$C_i = A_i B_i + B_i C_{i-1} + A_i C_{i-1}$
(dvakrat negiramo in dvonivojsko realiziramo)

12. realizacija skalarne funkcije z multipleksorji z enim vhodom.

-kaskadna realizacija funkcije z multipleksorji v več nivojih (kot nekakšna piramida)
Primer $x_1 x_2 + \overline{x_2} x_3 + \overline{x_1}\,\overline{x_2}\,\overline{x_3}$:



13. dokaz demorganovega zakona

Teorem T_{12}: $\overline{x+y} = \overline{x} \bullet \overline{y} \Rightarrow$ **1. De Morganov teorem**

Dokaz:

Da dokažemo gornji, t. i. 1. De Morganov teorem, bomo najprej dokazali, da velja:

$$(x+y)+\overline{x}\bullet\overline{y}=1$$

in da je: $$(x+y)\bullet\overline{x}\bullet\overline{y}=0$$

Zaradi 3. postulata moremo prvo enačbo razširiti takole:

$$(x+y)+\overline{x}\bullet\overline{y}=\underbrace{[(x+y)+\overline{x}]}_{1}\bullet\underbrace{[(x+y)+\overline{y}]}_{1}=1$$

Za desni del enačbe je očitno, da je enak 1, kar je bilo treba dokazati.
Za drugo enačbo velja razširitev:

$$(x+y)\bullet\overline{x}\bullet\overline{y}=\underbrace{[x\bullet\overline{x}\bullet\overline{y}]}_{0}+\underbrace{[y\bullet\overline{x}\bullet\overline{y}]}_{0}=0$$

Tudi tu zlahka spoznamo, da je desni del enak 0. če privzamemo:

$x+y=A$ in $\overline{x}\bullet\overline{y}=B$, smo torej ugotovili, da je:

$A + B = 1$ in $A \cdot B = 0$

Kadar za dve spremenljivki A in B veljata gornji enačbi, pravimo, da sta si komplementarni. Tedaj velja:

$\overline{A}=B$ oziroma $A=\overline{B}$

Z upoštevanjem dogovora oznak za A in B sledi 1. De Morganov teorem:

$\overline{x+y}=\overline{x}\bullet\overline{y}$ **ki smo ga s tem dokazali !!!**

14. enacbe vzbujanja registra

15. kritični prehodi (hazard)

Pojavi se zaradi zakasnitve pri spremembi vhoda iz enega nivoja v drugega. Če v vezju ni povratnih zank, po določenem času izhodne funkcije zavzamejo prave vrednosti. Asinhronsko vezje lahko zaradi teh napak preide v nepredvideno stanje tako, da ne moremo kontrolirati njegovega delovanja.

Statični hazard: je enkratna sprememba izhodnega signala prehodnega značaja, ko ne pričakujemo spremembe na izhodu. Nastopi v primeru, ko bi morala izhodna vrednost ostati enaka, kljub spremembi vhodov.

Dinamični hazard: do njega pride, kadar pričakujemo spremembo izhoda, ta pa se spremeni dvakrat; najprej pride do želene spremembe, nato se pojavi še začasni prehodni pojav z napačno vrednostjo.

16. zakaj se uporablja master/slave celica

Master/slave celico uporabljamo za sledilno vezje JK celice – spominska celica s predpomnjenjem.

17. narisu je en avtomat, pa je hotu vedt, če je lahko asinhronski

18. pragovne funkcije (tist k je na predavanjih razlagal - ravnina, eni cudni vektorji)

Preklopna funkcija je pragovna, če obstaja niz števil a1, a2,… - uteži in število P – prag tako, da velja:

$$f(x_1, x_2, \ldots x_n) = \begin{cases} 1; a_1 x_1 + a_2 x_2 + \ldots a_n x_n \geq P \\ 0; sicer \end{cases}$$

(P – hiperravnina, ki loči množico točk z vrednostjo 1 in 0)

19. simetricne fje (isto iz predavanj)

Funkcija je simetrična na spremenljivki x_i in x_j, če ju lahko zamenjamo, pri tem pa funkcijske vrednosti ostanejo enake.
Če je funkcija simetrična na vse pare spremenljivk je globalno simetrična funkcija.

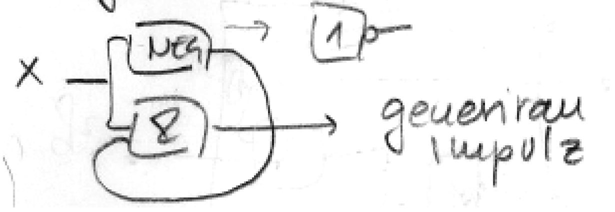
20. postulati, morš napisat v dualnih parih...

0*A=0	1+A=1
1*A=A	0+A=A
A*A=A	A+A=A
$\overline{A}$*A=0	$\overline{A}$+A=1

$$\overline{\overline{A}} = A$$
$$\overline{A * B} = \overline{A} + \overline{B}$$
$$\overline{A + B} = \overline{A} * \overline{B}$$

21. Proženje na pozitivno fronto. Kako bi generiral impulz iz tega (diracov)? In zakaj to sploh potrebuješ?

Rabiš en negator in ena AND vrata. Povežeš tako, da je X na vhodu neg. in vhodu AND vrat. Izhod negatorja je vezan na drug vhod AND vrat. Na izhodu je torej generiran impulz. Rabimo pa za CLOCK.



22. kje, v kakšnih vezjih potrebujemo URO/CLOCK.

Clock potrebujemo v sekvenčnih vezjih za sinhronizacijo spominskih celic, posledično prehajanje stanj.

23. Realizacija vezja NAND WIRED - prednosti in slabosti.

Kratkostičimo »žice«, pridobimo na delay-u, ki ga dejansko ni.

24. Razlika med sinhronskim in asinhronskim avomatom, kako se to vidi iz diagrama pregajanja stanj in pa neki v zvezi z uro.

Sinhronski avtomat potrebuje za delovanje uro. Asinhronski avtomat deluje hitreje – ni mu potrebno čakati na impulz ure, ampak pri spremembi vhoda takoj reagira. V diagramu prehajanja stanj to vidimo pri možnih kombinacijah, ki se lahko zgodijo v nekem stanju – spremeni se lahko le ena spremenljivka.

25. popolni (če ne gre, pa polovični) seštevalnik realiziran z dvovhodnimi NAND vsota je XOR, prenos AND, to se baje da realizirat z 4-mi NAND

popolni seštevalnik: vpr. št. 11

polovični:

$$S = \overline{A}B + A\overline{B} = A \oplus B$$

$$C = AB$$

26. Tipi prehodov stanj v asinhronskih avtomatih

-stabilni (direk v stabilno stanje)
-nestabilni (preko nestabilnega stanja v stabilno)

27. Funkcija avtomata (po izbiri)

vpr. št. 2

28. povezava med demultipleksorjem in dekodirnikom

Na podatkovni vhod demultipleksorja damo logično 1-ko in imamo dekodirnik.

29. Osnovna spominska enačba

$$Q(t+1) = g_1 Q(t) + g_2 \overline{Q(t)}$$

30. zapiši vhodne enačbe za register ki pomika levo desno in ima možnost vnos podatka serisko in paralelno

$$\Delta^1 Q_i = P x_i + S_R Q_{i+1} + S_L Q_{i-1}$$

(manka še serijski vnos!!)

31. Vzbujalne enacbe za serijski/paralelni pomikalni register

$$\Delta^1 Q_i = S_R Q_{i+1} + S_L Q_{i-1}$$

32. realizacija funkcije z NAND-WIRED-AND
33. T spominska celica (treba jo je narisat-dva AND in dva NOR elementa...)
34. Trivialne realizacije multipleksorja

Vse spremenljivke so pripeljane na adresne vhode multipleksorja, zato sta na podatkovne vhode pripeljani konstanti 0 in 1.

35. RS celica - moral sem narisati vezje z elementi, ter dokazati formulo RQ+S (R je prečno)