

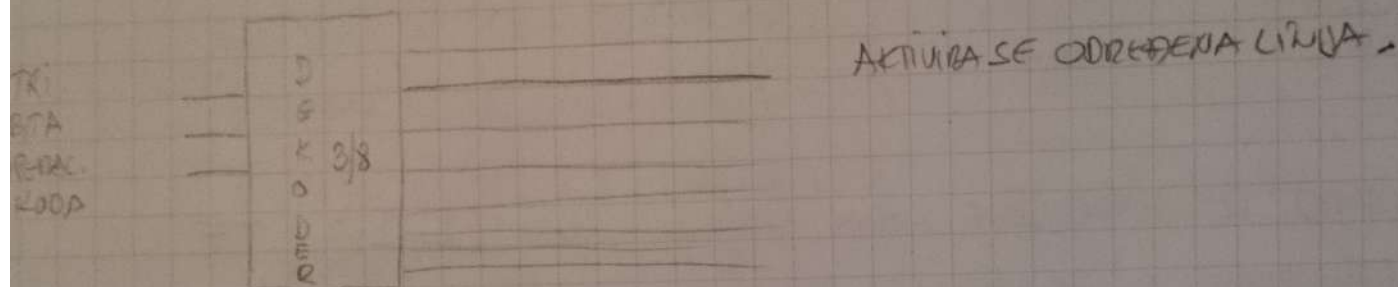
SKLOPONSKA IZVEDBA UPRAVLJANJA (RIBARJE POGLAVJE 7.3)

TEORIJA SAŽETAK

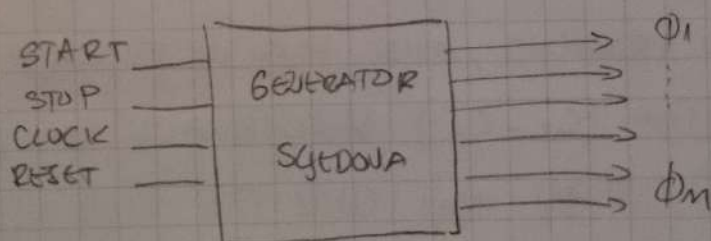
UPRAVLJAČKA JEDINICA, MOŽE BITI SKLOPONSKI IZVEDENA, PRILIČNO OLA
PREDSTAVLJA DVALE SEKVENCIJALNI SKLOP (AUTOMAT), KOJI GENERIŠA SJEĆOVI
UPRAVLJAČKIM SJEĆOVI.

RAZLIČNE RJEŠENJE SU TEŽKE (OSIM POTPUNIM REDIZAJNOM)

INSTRUKCIJSKI DEKODER TUMAČI OP. CODE, TE GENERIŠA ODSOUARAJUĆU LINIJU
ACTIVATION

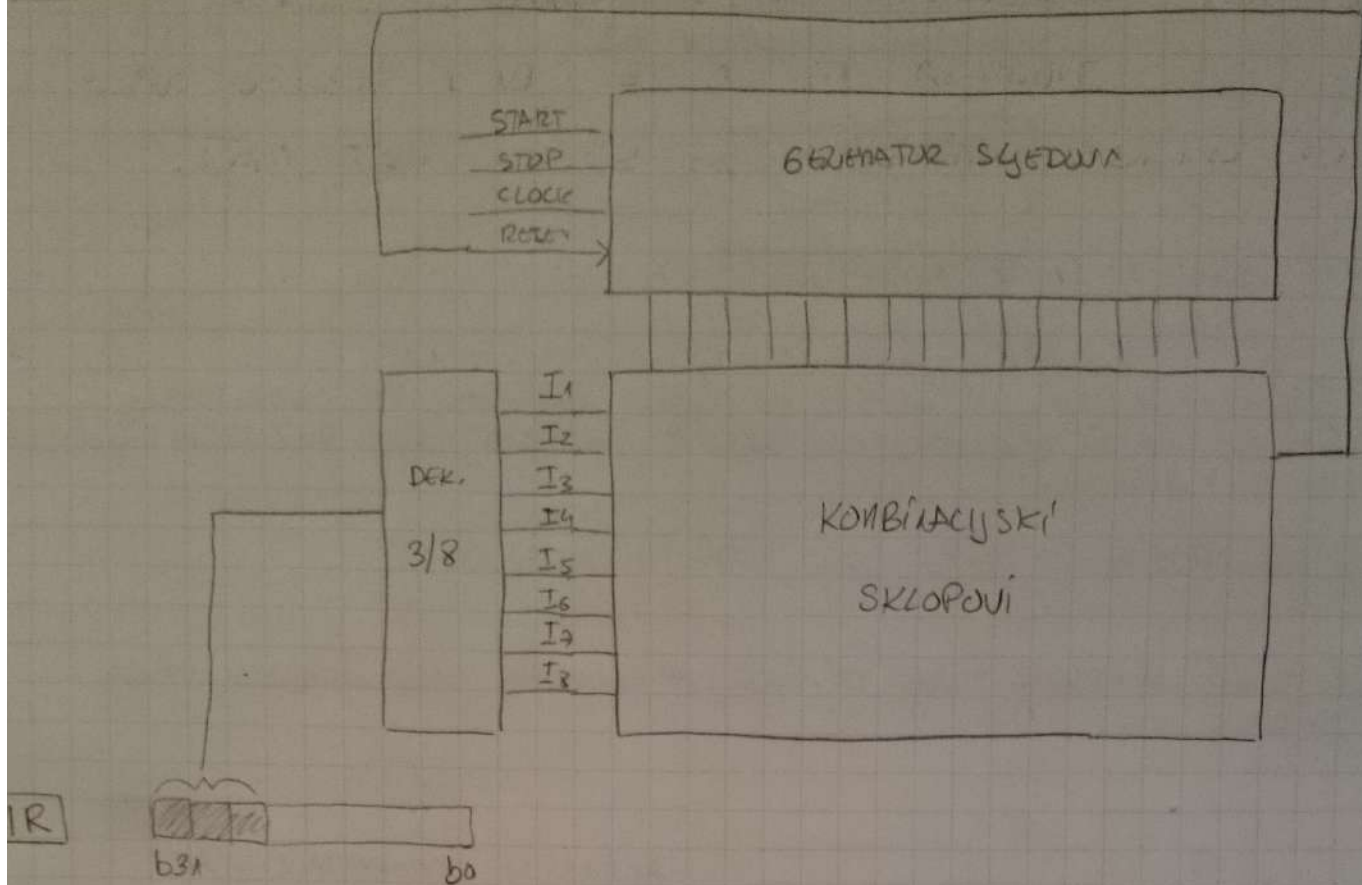


OKONITO LA TAJ DEKODER SPAJAJU SE IZLAZI IZ GENERATORA SJEĆOVI,
KOJI "DIKTIRA" KAKO ĆE SE INSTRUKCIJA IZVEDITI.



KOMBINACIJOM OVA DVA ELEMENTA OSTVARENA JE SKLOPONSKI LAŽA
UPRAVLJAČKA JEDINICA, ZA LAŽ NODER PROCESORNA!

UPRAVLJAČKA JEDINICA

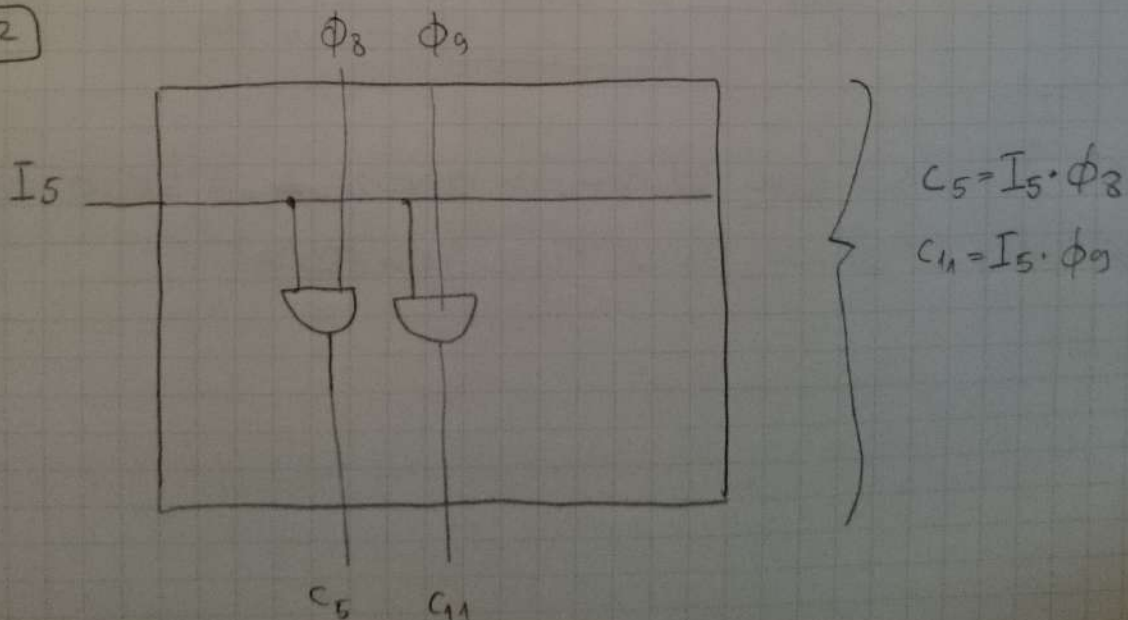


DAKLE IMAMO IZ OPERACIJSKOG KODA AKTIVNU LINIJU (I_1-I_8), TE IMAMO GENERATOR SLJEDOVA KOJI BROJI PO NODLU N SLJEDOVE.

⇒ MI SADA PROSPAJAJEM KOMBINACIJSKIH SKLOPOVA NA RAZNE NAČINE, MOŽEMO OSTVARITI RAZNE INSTRUKCIJE.

⇒ O TOME KAKO ĆEMO PROSPOJITI, OVISI KAKO JE FUNKCIJA INSTRUKCIJE OPISANA SA SIGNALIMA (C_1-C_x) U KOMBINACIJI SA ϕ_x

NAPRIMER



NAC MODEL (8-INSTRUKCIJSKI), IMA DEFINIRANE C_x SIGNALS, REGISTRE, PUT PODATAKA ITO \Rightarrow TAKO DA NI MOŽE MOĆNO UVRLO LAKO REKONSTRUIRATI JEDNAKOSTE SIGNALA INSTRUKCIJA!!

FAZA PRIBAVI JE ZA VAŽ NODS UVIJEK ISTA ZA SVE INSTRUKCIJE

1. $PC \rightarrow MAR$ // STAVI PC U ADRESNI REG
2. $M[MAR] \rightarrow MDR$ // DOHVAĆI SA TE ADRESE, U MDR
3. $PC = PC + 1$ // POVEĆAJ PC
4. $MDR \rightarrow IR$ // STAVI DOHVAĆENO U IR

DETALJNIJI POKAZ FAZE PRIBAVI

- | | | | |
|-----------|------------------------------|---|---------------------------|
| 1. EPC : | $C_8 = I_x(\phi_1 + \phi_2)$ | } | $PC \rightarrow MAR$ |
| 2. LMAR : | $C_{13} = I_x(\phi_2)$ | | |
| 3. READ : | $C_3 = I_x(\phi_3 + \phi_4)$ | } | $M[LMAR] \rightarrow MDR$ |
| 4. LMDR : | $C_{14} = I_x(\phi_4)$ | | |
| 5. INC : | $C_{16} = I_x(\phi_5)$ | } | $PC++$ |
| 6. ENDR : | $C_5 = I_x(\phi_6 + \phi_7)$ | | |
| 7. LIR : | $C_{12} = I_x(\phi_7)$ | } | $MDR \rightarrow IR$ |

ϕ_7 DEKODIRANJE OP. KODA

I_x = JEDNOST MOŽE BITI BILU KOJA INSTR ($I_1 - I_8$)

PRAVILA :	
ENABLE	TRAJE 2ϕ
READ	TRAJE 2ϕ
WRITE	TRAJE 2ϕ
LOAD	TRAJE 1ϕ
ALU OP.	TRAJE 2ϕ