

Digitalna vezja, 2020/2021

Seminar: Kalkulator

Pripravl/a: Ime in priimek, vpisna številka

Datum: _____

1. Opis problema

Tu naj bo na kratek in razumljiv način predstavljen problem, ki ga rešujemo.

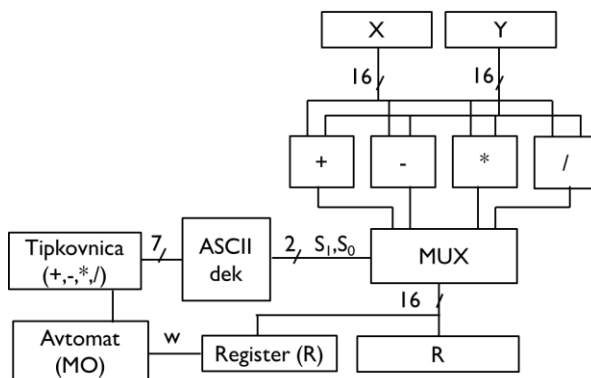
Realizacija enostavnega kalkulatorja kot Mooreov avtomat, ki ima naslednje zahteve:

- Vhod: vpis 16-bitnih dvojiških števil $X=(x_{15}, x_{14}, \dots, x_1, x_0)$, $Y=(y_{15}, y_{14}, \dots, y_1, y_0)$.
- Aritmetične operacije, ki se izvedejo nad podatki X in Y: seštevanje(+), odštevanje (-), množenje (*), deljenje (/).
- Shranjevanje rezultata $R=(r_{15}, r_{14}, \dots, r_1, r_0)$ v register.
- Prikaz vhodnih podatkov (X, Y) in rezultata (R).

2. Blok shema z opisom predlagane rešitve

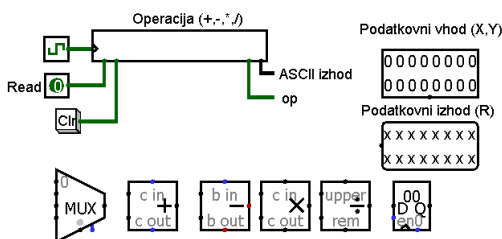
Tu naj bo shema vezja, kjer so predstavljeni osnovni gradniki, ki so med seboj povezani za implementacijo. Za posamezne povezave naj bo podano število signalov, ki predstavljajo podatke ali krmilne vhode. Podani so obstoječi gradniki v logisimu in opisani moduli, ki jih je potrebno razviti.

Blok shema digitalnega vezja, ki deluje kot kalkulator je prikazana na sliki 1.



Slika 1: Blok shema kalkulatorja

Za realizacijo kalkulatorja bomo uporabili gradnike, ki so na voljo v logisimu (slika 2).

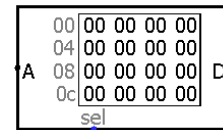


Slika 2: Logisim: Tipkovnica, Podatkovni vhod, aritmetični gradniki, MUX, register.

Definirati moramo tudi ASCII dekodirnik za aritmetične operacije. Uporabili bomo ROM za realizacijo krmilnih signalov S_1 in S_0 pri 2-naslovnem MUXu.

Operacija	ASCII	S_1	S_0
+	0101011	0	0
-	0101101	0	1
*	0101010	1	0
/	0101111	1	1

Naslov (A) _H	Podatek (D) _H
2B	0
2D	1
2A	2
2F	3

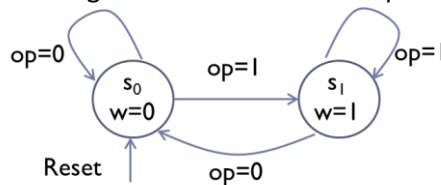


ROM:

3. Opis avtomata - DPS

Tu naj bo predstavljen Mooreov avtomat s stanji, ki določajo izhode in prehode med stanji, tako da je ponazorjeno delovanje celotnega sistema.

- 1-Signal na izhodu tipkovnice $op = 1$ določa, da je operacija (+, -, *, /) vpisana na tipkovnici.
- 2-Izhod $w = 1$ omogoči vpis rezultata v register R na izhodu MUXa po izvedeni aritmetični operaciji.



4. Aplikacijska tabela za krmiljenje posameznih modulov (vhodna logika, časovne spremenljivke-register stanj, izhodna logika).

op	Q(t)	D Q(t+1)	w
0	0	0	0
0	1	0	1
1	0	1	0
1	1	1	1

Register stanj realiziramo z D pomnilno celico: vhod $D = op$; izhod $w = Q(t)$.

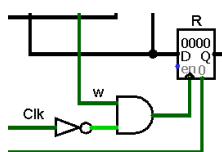
5. Realizacija vezij, ki niso predstavljena v točki 2 in točki 4.

Tu naj bodo opisane rešitve, ki so v logisimu predstavljene kot

- *podvezje, ki izvede krmiljenje določenega gradnika,*
- *dodatno vezje logičnih vrat, ki krmili modul ali gradnik,*
- *izvedba primerjave, kodiranja ali dekodiranja signalov,*
- *vnos ali pretvorba podatkov za določeno operacijo.*

ASCII koda je naslov, ki določa izbrano operacijo v ROMu. Shrani se v register Reg A ob pozitivni fronti urinega signala Clk.

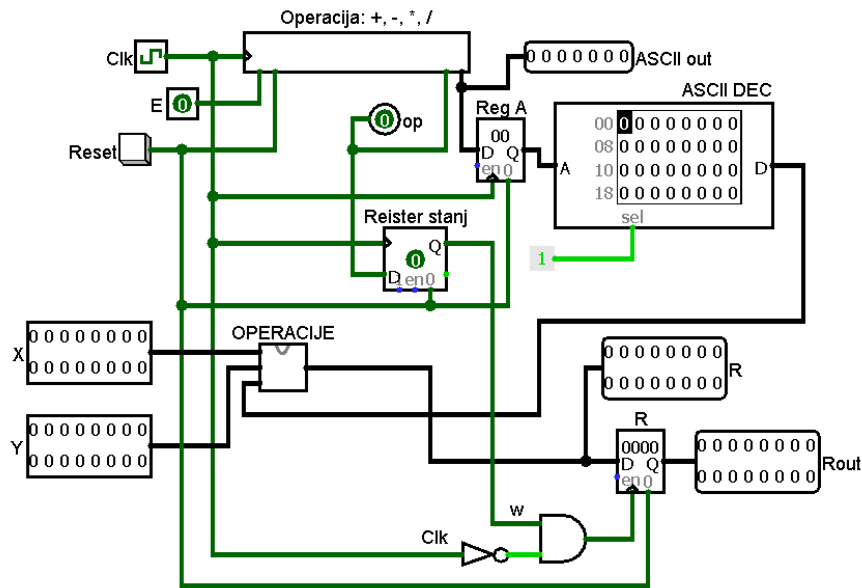
Signal w za vpis v rezultata register R omogoča vpis v drugi polovici periode (negativna fronta), zato je urin signal določen kot $(\overline{clk}. w)$.



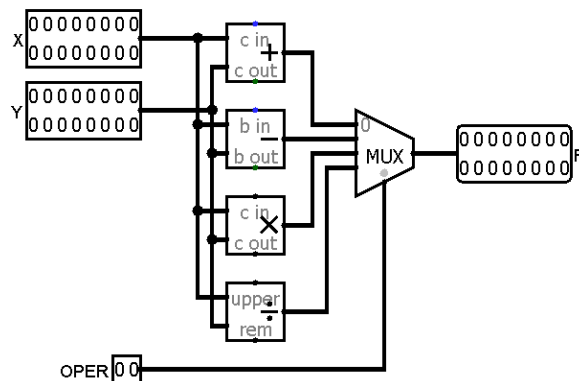
6. Logična shema vezja za realizacijo kalkulatorja v logisimu.

Tu naj bo podano glavno vezje in nato še vsa podvezja z imeni, ki so uporabljena v logisimu.

Glavno vezje kalkulatorja:



Vezje za izvedbo aritmetičnih operacij - OPERACIJE



7. Testiranje kalkulatorja.

- Na tipkovnici vpišemo aritmetične operacije +, -, *, /,
- V vsaki urini periodi se na izhodu MUXa prikaže rezultat operacije in se shrani v register R. Primeri izračunov:

Operacija	X	Y	R
X + Y	3	2	5
X - Y	3	2	1
X * Y	3	2	6
X / Y	3	2	1

Pri testiranju niso vključene zastavice N,Z,V,C, ampak samo izvedba aritmetičnih operacij +, -, *, / z gradniki logisima.