

# Elektronika za začetnike - CMOS logična vezja serije 4000 II (15)

Avtor: Bojan Kovač

E-pošta: bojan@svet-el.si

*Logična vezja so razdeljena na različna področja s svojimi vnaprej določenimi funkcijami. V praksi se je izkazalo, da v vezjih vedno potrebujemo neke funkcije, na primer prikaz na 7-segmentnem LED prikazovalniku, števce, premikalne registre in podobno, zato so proizvajalci vsako takšno funkcijo združili v enem čipu, ki ga moramo le še povezati z drugimi funkcijami, da opravlja svojo nalogo. Prav povezovanje različnih funkcij v celoto je tudi osnova vsakega programiranja mikrokontrolerjev, da opravlja svojo nalogo tako, kot smo si želeli.*

## ELEKTRONIKA ZA ZAČETNIKE



Izbor funkcij je prepuščen iznajdljivosti vsakega posameznika in tudi medsebojne povezave med čipi se večkrat razlikujejo, čeprav izvajajo popolnoma enake naloge. Tudi v tem je čar elektronike, da lahko enake naloge izvajamo na različne načine in lahko tako uporabimo vso svojo kreativnost in iznajdljivost tako, da ima končni izdelek čim manj elementov, čim manjšo površino in posledično tudi čim nižjo ceno!

## FUNKCIONALNA RAZDELI-

### TEV LOGIČNIH VEZIJ

Na grobo lahko CMOS logično družino razdelimo na:

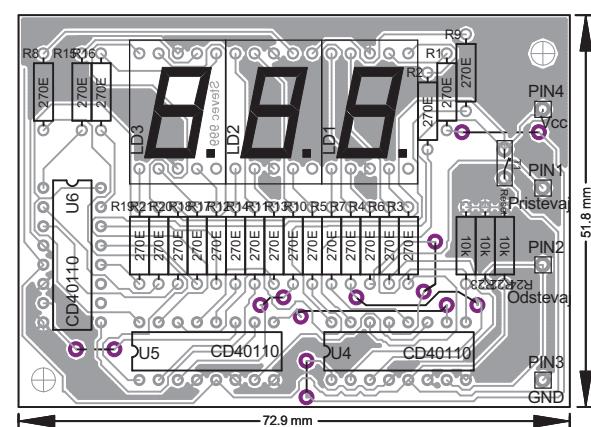
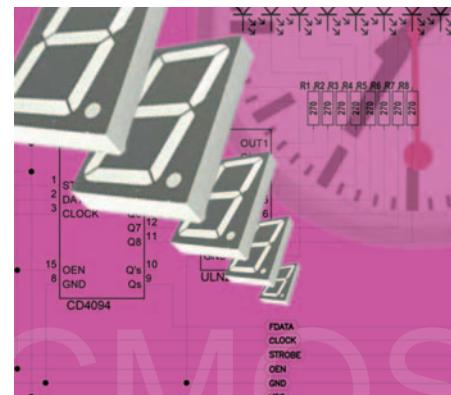
- » logična vrata,
- » števci,
- » enkoderje in dekoderje,
- » flip-flope in zapuhe (latch),
- » multipleksanje in demultipleksanje,
- » linijske ojačevalnike in gonilnike,
- » premikalne registre,
- » aritmetične logične enote (ALU) in
- » vezja s posebnimi funkcijami.

Poleg čipov, ki vsebujejo posamezna logična vrata iste ali različnih vrst, so vsi ostali kombinacija medsebojnih

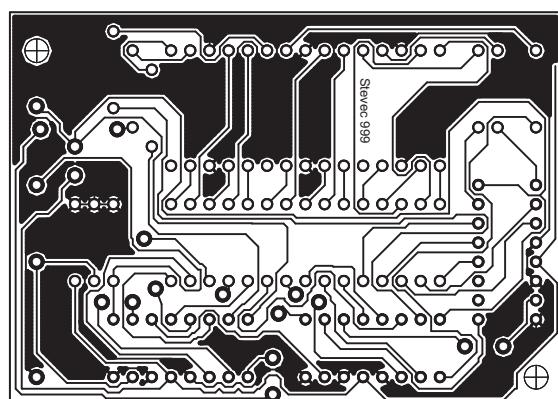
povezav več sto logičnih vrat v funkcionalno celoto.

Števci štejejo vhodne impulze na različne načine, sinhrono in asinhrono ter se z drugimi števcji lahko povezujejo v kaskade, kar pomeni, da lahko za štetje večjih vrednosti enega za drugim dodajamo poljubno število števcov. Praktičen primer števca, ki lahko trenutni vrednosti en impulz prišteje ali odšteje vidimo na sliki 1. Števec lahko razširimo na poljubno število mest, pri narisani shemi pa lahko šteje do 999. Eden izmed dobrih primerov uporabe takšnega števca bi bil za štetje parkiranih avtomobilov v posameznih etažah javnih garaž. Ko avtomobil pripelje v etažo, se število, ki predstavlja parkirane automobile poveča, ko pa eden od avtomobilov zapusti etažo, se to število zmanjša.

Obstajajo BCD in binarni števci. BCD števci so takšni, ki štejejo le od 0 do 9,



Slika 2: Montažna ploščica TIV števca od 0 do 999, dimenzije 72 x 52 mm.

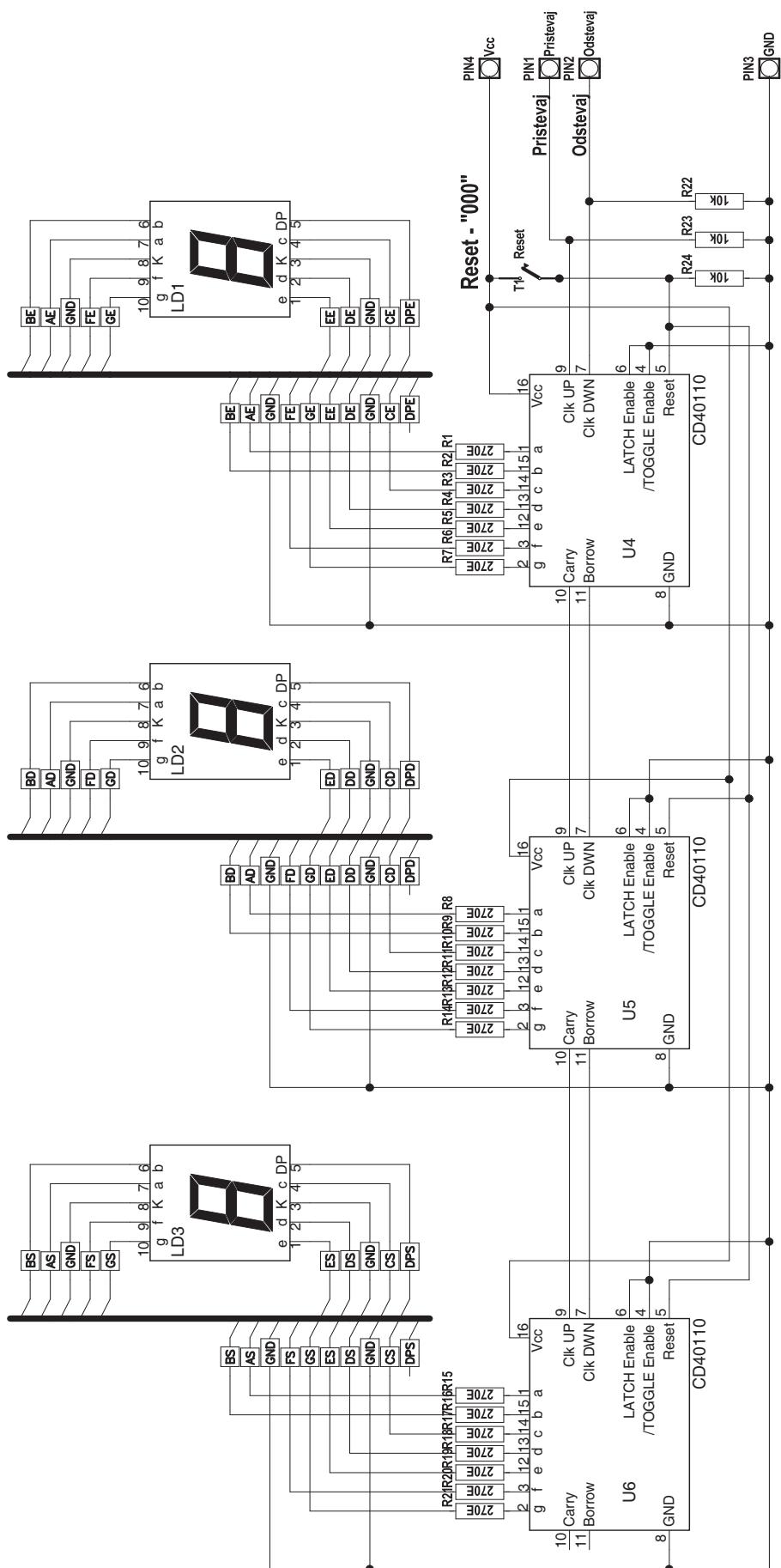


Slika 3: Tiskano vezje števca.

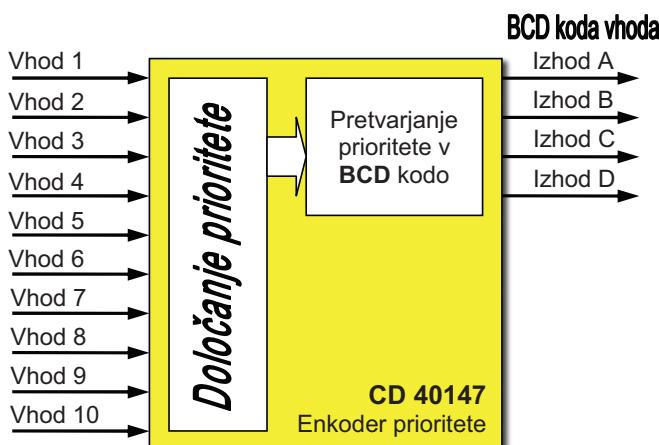
čeprav imajo 4 vhode, binarni pa štejejo v binarnem načinu, torej od 0 do 15 (od 0000 do 1111 binarno). Če povežemo po dva enaka števca, bosta BCD štela od 0 do 99, binarna pa od 0 do 255 (od 00000000 do 11111111 binarno).

Z enkoderji lahko poljuben nabor stanj vhodnih signalov s kombinacijo osnovnih logičnih vrat spremenimo v neko kodo. Tipičen primer takšnega enkoderja je vezje 40147, enkoder prioritete, ki daje na izhodu binarno kodo v odvisnosti od tega, kateri od desetih vhodov je postavljen. Vhod številka 10 ima najvišjo prioriteto. Ne glede na to, kateri vhod je še na HI nivoju, bo na izhodu prikazana binarna koda tistega vhoda s HI nivojem, ki ima višjo prioriteto. Predstavljajmo si, da imamo deset različnih virov zunanjih prekinitev in le en prekinitveni vhod na mikrokontrolerju. Vseh deset vhodov prek ALI vrat povežemo na ta prekinitveni vhod, istočasno pa preberemo tudi 4-bitno binarno kodo, ki nam pove, kateri vir je povzročil prekinitvev. S petimi vhodi na mikrokontrolerju lahko torej zaznamo prekinitvev in v prekinitveni rutini preberemo, kateri od desetih virov je povzročil prekinitvev. Glede na vir prekinitve potem izvedemo ustrezno rutino. Blok shema takšnega vezja je na sliki 4 skupaj z resničnostno tabelo. Znaki »x« v resničnostni tabeli pomenijo, da vrednost vhoda na tistem mestu ni pomembna in ne vpliva na stanje na izhodu.

Tipičen primer dekoderja so BCD števci, ki imajo že vgrajen dekoder in gonilnik za 7-segmentni LED prikazovalnik. Binarna koda se dekodira v priziganje ustreznih segmentov LED prikazovalnika. Druga možnost za uporabo dekoderja je spet v kombinaciji z mikrokontrolerjem. Z neko 4-bitno kodo na vhodu dekoderja 4028 lahko naslavljamo 10 različnih integriranih vezij, na primer RAM spomin. Vsak izhod povežemo z vhodom CE (chip enable, omogoči integrirano vezje) drugega vezja in na ta način izbiramo, s katerim želimo delati – brati ali vpisati podatke. Ta primer vidimo na sliki 5. Če nam zadostuje le 8 izhodov, povežemo vhod D na LO nivo in za naslavljjanje priključimo le tri izhode mikrokontrolerja.



*Slika 1: Števec, 7-segmentni dekoder in gonilnik 7-segmentnega LED prikazovalnika v enem samem CMOS čipu CD40110.*



Vhodi										Izhodi			
1	2	3	4	5	6	7	8	9	10	D	C	B	A
0	0	0	0	0	0	0	0	0	0	1	1	1	1
1	0	0	0	0	0	0	0	0	0	0	0	0	0
X	1	0	0	0	0	0	0	0	0	0	0	0	1
X	X	1	0	0	0	0	0	0	0	0	0	0	1
X	X	X	1	0	0	0	0	0	0	0	0	1	1
X	X	X	X	1	0	0	0	0	0	0	1	0	0
X	X	X	X	X	1	0	0	0	0	0	1	0	1
X	X	X	X	X	X	1	0	0	0	0	1	1	0
X	X	X	X	X	X	X	1	0	0	0	1	1	1
X	X	X	X	X	X	X	X	1	0	1	0	0	0
X	X	X	X	X	X	X	X	X	1	1	0	0	1

Slika 4: Obravnavanje desetih različnih virov prekinitve glede na prioriteto.

Flip-flopi so vezja s sposobnostjo pomnenja 1 bita informacije. Primerjamo jih lahko z bistabilnim multivibratorjem, ki ima prav tako dve stabilni stanji in sposobnost pomnjenja enega bita informacije. Z logičnim nivojem HI na vhodu »S« (Set) flip-flopa izhod »Q« postavimo na HI logični nivo. Ta nivo ostane na izhodu toliko časa, dokler ga s HI nivojem na vhodu »R« (Reset) ne postavimo na LO nivo. Izhod »Q« (Q prečno, Q negirano) se vedno postavi v nasprotno logično stanje kot izhod »Q«.

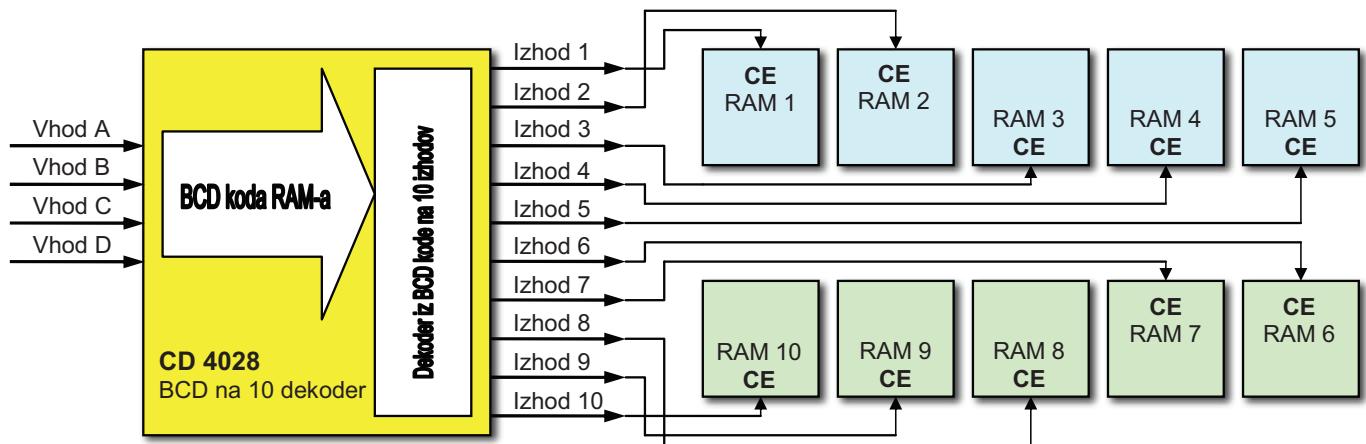
Zapah (latch) prepiše stanje na svojem izhodu, vendar ravno v trenutku, ko to omogočimo s HI nivojem na priključku E (enable) in za toliko časa, dokler vsebine ne prepišemo s stanjem, ki se v trenutku vpisa nahaja na vhodu. Imamo različne vrste zapahov: RS, D in T. RS je Set/Reset zapah, D je Data (podatkovni) zapah in T je Toggle (preklo-

pni) zapah. Največkrat uporabljamo D zapah, ki na izhod prepiše stanje na svojem »D« vhodu. Za vhod je uporabljen le en priključek. D zapahi so sestavni del premikalnih registrov. Na na D vhod postavimo nek podatek, na EN (enable, omogočeno) vhod pa preklopimo takt iz LO na HI in nazaj na LO logični nivo. S tem je podatek vpisan v zapah in ga lahko preberemo na Q izhodu. Ta izhod povežemo na D vhod naslednjega zapaha.

Multiplekserji so elektronski preklopniki, s katerimi lahko v vsakem trenutku izmed več mogočih virov vhodnega signala z naslavljanjem izberemo in obdelujemo le enega. Demultiplekserji so preklopniki, s katerim lahko nek signal z naslavljanjem pripeljemo na enega od več različnih izhodov, ki jih imamo na razpolago. Naslavljanje je običajno binarno. Če si multiplekser in demultiplekser predstavljamo po-

vezana, je med njima le ena signalna povezava, kljub temu pa lahko signal iz kateregakoli vhoda povežemo s katerimkoli izhodom. Tudi v samem mikrokontrolerju je že vgrajen multiplekser, s katerim izbiramo vhod, na katerem bomo merili signal z AD pretvorno. Seveda bi bilo nesmiselno imeti osem analogno-digitalnih pretvornikov na mikrokontrolerju z osmimi analognimi vhodi, zato zaporedoma povezujemo posamezne vhode z AD pretvornikom in vsakemu zase izmerimo napetost. Shematsko je delovanje in upravljanje kombinacije multiplekserja in demultiplekserja prikazana na sliki 8. Ustrezno povezavo med enim izmed vhodov in enim izmed izhodov določimo s naslovnimi priključki, ki jih upravljamo z mikrokontrolerjem.

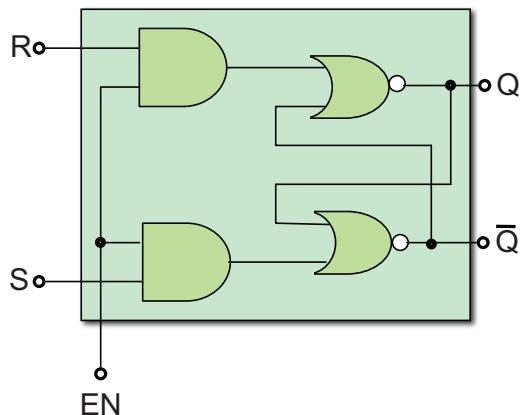
Linijske ojačevalnike in gonilnike uporabljamo za povečanje tokovne ali napetostne zmogljivosti nekega vodila ali



Slika 5: Dekoder iz BCD na 10 izhodov.

## SAMOGRADNJE

RS flip-flop z omogočanjem vpisa



Slika 6: RS flip-flop in RS zapora (latch).

signalna, prilagoditvi signalna na drugačne napetostne nivoje ali upravljanja z večimi priključenimi linijami. Tipični primer sta CD4010 in CD4009, ki logični nivo vhodnega signala tudi obrne (invertira).

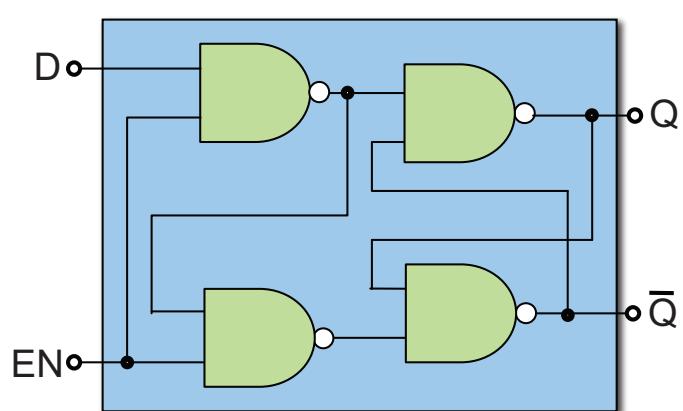
Premikalni registri imajo to lastnost, da v njih informacija potuje od prvega do zadnjega flip-flopa. So idealni za pretvarjanje nekega binarnega podatka iz paralelne v serijsko obliko in obratno. Poleg tega so lahko ti registri še serijsko-serijski, ki informacijo za določen čas zadržijo, saj se informacija med flip flopi premika počasneje, kot se spreminja vrednost 1-bitne informacije v serijski obliku na vhodu. Lahko jim rečemo, da so neke vrste digitalni zakasnilni element.

CD4094 je premikalni register, kateremu podatke vpisujemo v serijski obliku, na izhodnih priključkih pa jih v trenutku vpisa (STROBE=HI) dobimo

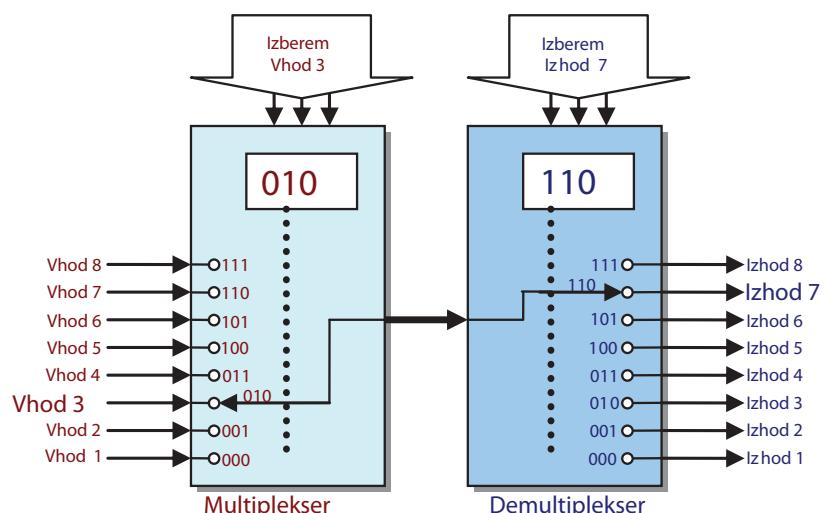
v paralelni obliki. To integrirano vezje nam omogoča tudi to, da lahko izhod postavimo v stanje visoke impedance s HI logičnim nivojem na krmilnem priključku OE (output enable, izhod omogočen).

Lep primer uporabe takšnega premikalnega registra, s katerim lahko po želji razširimo izhodne zmogljivosti mikrokontrolerja na MiniPinu, je kasadna vezava ustreznega števila premikalnih registrov CD4094. Vsi vemo,

D zapah (latch)



Slika 7: D zapah prenese stanje na vhodu »D« na izhod »Q«.



Slika 8: Povezava multiplekserja in demultiplekserja.

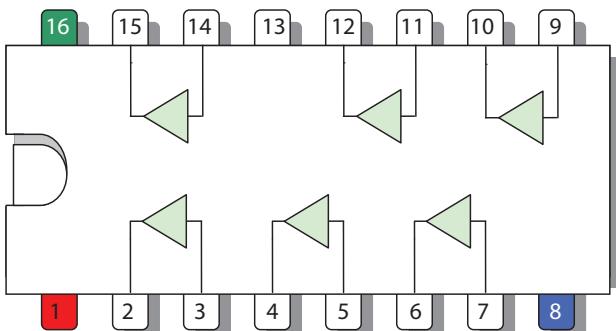


**BEEPROG+** JE UNIVERZALNI PROGRAMATOR  
NASLEDNJE GENERACIJE, KI SE MU NADGRAJUJE  
LE PROGRAMSKA OPREMA IN DODAJA PODPORA  
NOVIM KOMPONENTAM. POSODOBITVE SO  
BREZPLAČNE IN SPROTI SLEDIJO  
DOGAJANJU NA PODROČJU  
PROGRAMABILNIH ELEKTRONSKIH  
VEZIJ.

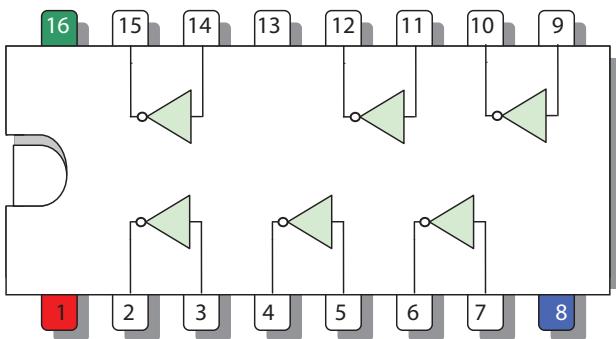
**BeeProg+**



CD4010 - 6 x linijski ojačevalnik

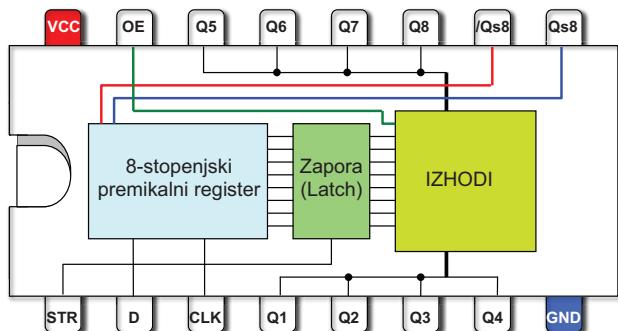


CD4009 - 6 x linijski ojačevalnik - inverter



Slika 9: Invertirani CD4009 in običajni linijski ojačevalnik CD4010.

CD4094 - premikalni (shift) register

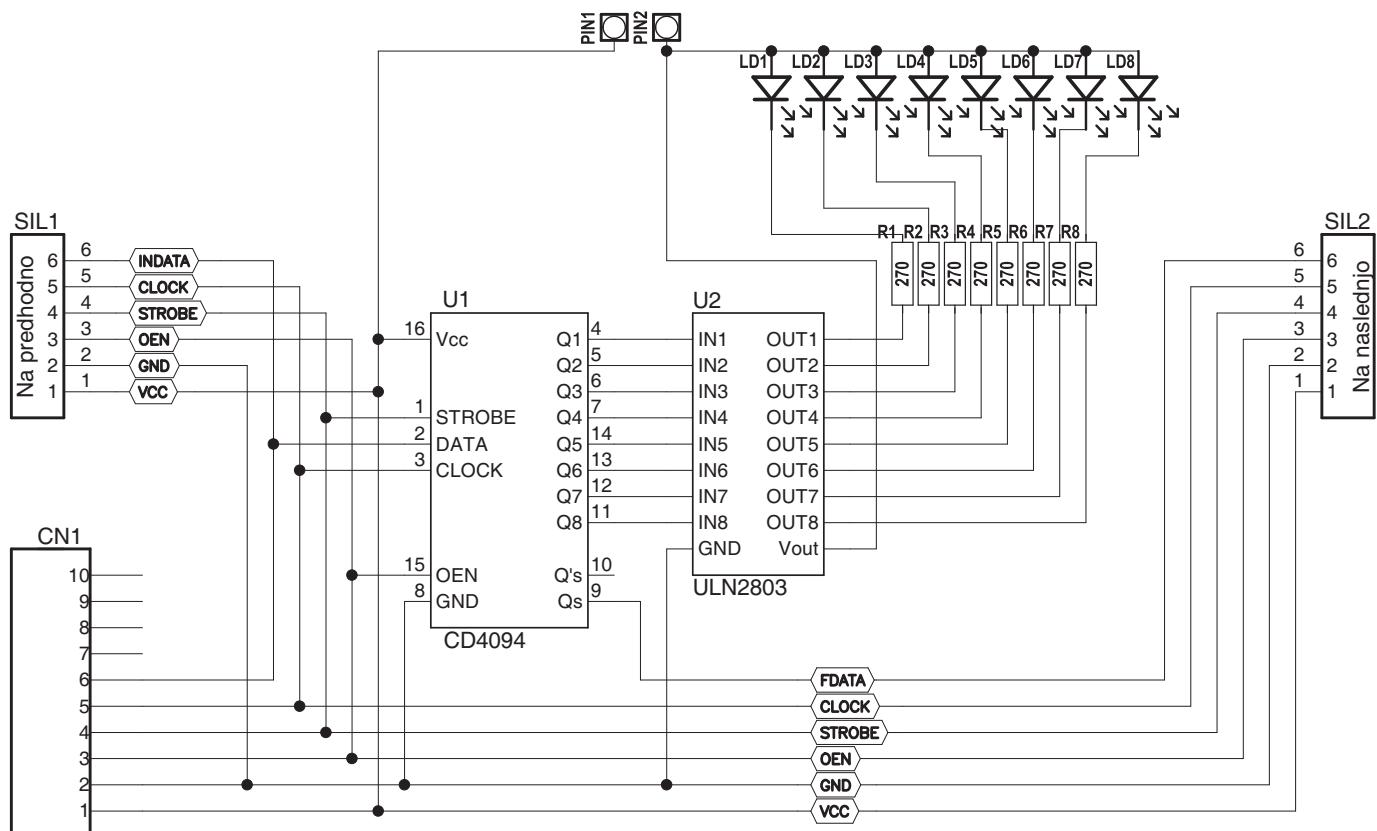


Slika 10: Notranjost premikalnega registra CD4094.

da so štiri izhodna vrata s po osmimi priključki včasih pre malo za vse, kar bi želeli postoriti z mikrokontrolerjem. Če v svojem projektu uporabimo celo katerega od 20-pinskih ali celo 8-pinskega, pa smo včasih res v stiski, kar se tiče iz hodov. Predlagam preprost in zelo uporaben dodatek, pri katerem bomo s samo štirimi kontrolnimi priključki upravljali s celo verigo izhodov, ki jo bomo lahko poljubno razširili! Shema tega vezja je na sliki 11

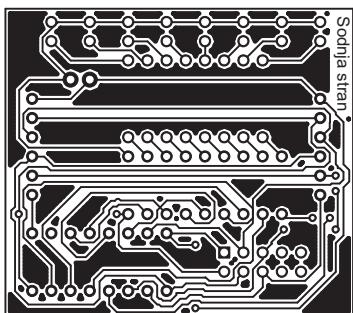
Tiskano vezje in montažni mačrt tega dodatka za MiniPin ali MegaPin razvojno ploščico lahko vidite na sliki 12.

Podatek postavimo na DATA linijo in dvignemo in spustimo CLOCK linijo. S tem podatek vpšemo v prvi flip-flop.



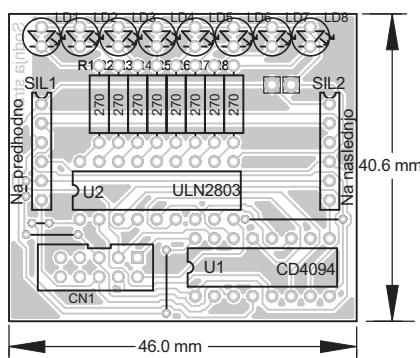
Slika 11: Shema izhodne razširitve za Minipin/Megapin.

# SAMOGRADNJE



**Slika 12:** Ploščica tiskanega vezja izhodne razširitve.

Informacija se skupaj s taktom pomika proti zadnjemu flip-flopu, od tam pa še naprej na naslednje stopnje, če imamo priključene. Podatek, ki smo ga najprej postavili na podatkovno linijo, se bo premaknil na najbolj oddaljeno lokacijo, zato moramo to v protokolu pošiljanja upoštevati. Podatke, ki jih želimo prenesti na paralelne izhode serijsko pomikamo od prvega do zadnjega flip-flopa. To premikanje informacije ne vpliva na stanje izhodov, dokler tega ne želimo. Iz premikalnih registrov lahko vsebino v katerem koli trenutku prepišemo v izhodne zapore (latch) s HI nivojem na priključku STROBE. Logični nivoji vseh premikalnih registrov se istočasno prepišejo v izhodne registre (zpora, latch), ki to informacijo zadržijo do naslednjega vpisa. Zaporeno lahko vežemo večje število takšnih razširitvenih vezij in vsak nam bo povečal število izhodnih priključkov za osem! Prvo razširitveno vezje povežemo z MiniPin prek CN1, naslednja vezja pa dodajamo na desni



**Slika 13:** Montažni načrt.

strani in jih povežemo s konektorjem SIL2 predhodne in SIL1 naslednje. Na naslednjih stopnjah konektor CN1 seveda ni potreben in ga ne prisppajkamo. S pinoma Pin1 in Pin2 na breme pripeljemo napetost 5 V iz MiniPina. Lahko pa na PIN2 pripeljemo tudi kakšno drugo napajalno napetost, ki je lahko tudi višja od 5 V, saj ULN2803 lahko deluje tudi pri izhodnih napetostih do 50 V, ob tem, da krmilna napetost za krmiljenje izhodnih tranzistorjev (ki delujejo kot stikala) še vedno ostaja na logičnem nivoju. Ploščica je majhna in za potrebe prikaza uporablja le LEDice, vendar bi lahko namesto njih poveza-

li tudi releje, žarnice, kakšen motor in podobno.

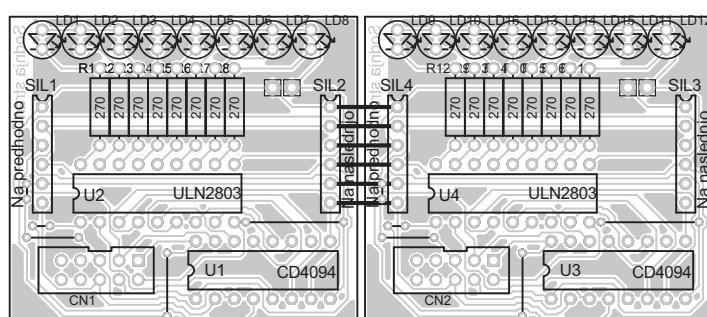
Aritmetične logične enote (ALU) izvajajo logične aritmetične operacije med dvema številoma. Lahko gre samo za primerjavo, katero je večje, manjše ali če sta enaki, lahko pa se izvajajo tudi seštevanje ali odštevanje nakar se primerja rezultat.

Vse te funkcije lahko danes nadomeščimo s programom znotraj mikrokontrolerja, ki nadomešča desettisoče logičnih vrat, poleg tega pa mu logične funkcije lahko spremojamo s programom in isto integrirano vezje uporabimo v čisto drugačnem vezju s čisto drugačno funkcijo. S tem smo, gledano v svetovnem merilu, zmanjšali število različnih komponent na tržišču, zmanjšali potrebne svetovne zaloge teh komponent, zmanjšali velikost tiskanih vezij za vgradnjo funkcijsko enake elektronike, omogočili ponovno uporabo istih vezij za povsem drugačne funkcije s preprosto spremembo programa in zmanjšali porabo energije.

## ZAKLJUČEK

Zaradi vseh zaslug ter pionirskega dela in sodelovanja na področju uvajanja prvih mikrokontrolerjev in podpore v obliki prve mikroprocesorske periferije lahko tej družini priznamo, da je ena od velikih legend, ki so krogile trende razvoja v elektroniki in da bi bil brez nje danes svet elektronike in tudi Svet elektronike drugačen kot je.

[www.svet-el.si](http://www.svet-el.si)



**Slika 14:** Kaskadna vezava dveh razširitvenih vezij.

**AX ELEKTRONIKA**  
GENERALNI ZASTOPNIK  
ZA SLOVENIJO

**AX ELEKTRONIKA D.O.O.**  
POT HEROJA TRTNIKA 45,  
1000 LJUBLJANA  
TEL.: 01 549 14 00  
INTERNET: [WWW.SVET-EL.SI](http://WWW.SVET-EL.SI),  
E-MAIL: [STIK@SVET-EL.SI](mailto:STIK@SVET-EL.SI)

**SMARTPROG2 JE VSESTRANSKI 40-PINSKI PROGRAMATOR Z ISP IZBIRO IN Z USB POVEZAVO NA PC, KVALITETEN HARDVER JE NADPOLNjen Z VRHUNSKIM PROGRAMOM, KI KOT STANDARDE PODPira VSE MS Windows OPERACIJSKE SISTEME (OD WIN.95 DO WIN7). PROGRAMIRA VEČ KOT 24100 RAZLIČNIH MIKROKONTROLERJEV.**

**SmartProg2**  
universal 40-pindrive programmer

