

8. ARHITEKTURA U/I I JEDNOSTAVNI U/I PORTOVI KOD FAMILIJE MIKROPROCESORA MC68xxx

8.1. Izolovani i memorijski preslikani U/I

Postoje dva osnovna načina za realizaciju ulaza/izlaza kod mikroracunara. Kod prvog od njih, izolovanog ulaza/izlaza, čitanje U/I portova i upis u njih vrši se posebnim ulaznim, odnosno izlaznim operacijama. Na primer, instrukcijom IN 5 se reč iz U/I porta 5 prenosi u registar podataka, dok se instrukcijom OUT 3 reč iz registra podataka prenosi u treći U/I port. Kod drugog načina, memorijski preslikanog ulaza/izlaza, U/I portovi se smatraju lokacijama u memoriji, tako da se čitanje i upis vrši istim naredbama kao i čitanje i upis u glavnu memoriju. Ako je, na primer, adresa jednog takvog porta \$4000, tada će instrukcija MOVE \$4000,D0 predstavljati ulaznu, a MOVE D0,\$4000, izlaznu instrukciju. U ovom slučaju nema potrebe za posebnim U/I naredbama, pa neke arhitekture, kao što je familija MC68000, koriste isključivo memorijsko preslikani U/I, dok neke druge imaju mogućnost korišćenja obe U/I tehnike.

Svaka od prethodno pomenutih tehnika ima svoje prednosti. Instrukcije izolovanog U/I, kao kod 8-bitnog mikroprocesora INTEL 8080, koriste jednobajtni opkod i jednobajtnu adresu porta, dok većina instrukcija memorijski preslikanog U/I 8-bitnih mikroracunara obično ima dužinu od tri bajta ili su spregnute sa adresnim registrom pa su dvobajtne. Ova dodatna dužina instrukcija može biti ozbiljan problem kada neki aplikacioni program koristi puno U/I operacija a mora se smestiti u što je moguće manji deo memorije. Međutim, familija MC68000 ima adresne načine rada, kao što je adresno registarsko indirektno adresiranje, koji se mogu upotrebiti da bi se poboljšala statička efikasnost U/I rutina. Adresni dekodler je jednostavnije rešenje, jer je potrebno analizirati svega 8 adresnih bitova, a ne 16 ili više, uz neke upravljačke signale. Najvažnije je da izolovani U/I nije u istoj meri osetljiv na softverske greške, kao što je to memorijsko preslikani U/I.

Ipak, memorijsko preslikani U/I je popularniji zato što većina mikroracunara ima naredbe koje direktno izvršavaju u memoriji, kao što su ADDQ #1,\$4000, ROL \$4000, ili instrukcije tipa registar→memorija kao što je ADD D0,\$4000. Ako se program nalazi u ROM, može se upotrebiti indeksirano adresiranje za relokaciju memorijsko preslikanog U/I, dok kod izolovanog U/I takva mogućnost ne postoji.

Treba biti svestan mogućnosti slučajnog upisa u izlazne portove kada su koristi memorijsko preslikani U/I. Međutim, moguće je izvršiti zaštitu korišćenjem tzv. brave (lock). Brava predstavlja izlazni port koji sam nije "zabavljen" pa program može da vrši izmenu njegovog sadržaja. Vršiti se I operacija nad svim bitovima sadržaja brave i adresnim bitovima i drugim upravljačkim signalima, kako bi se dobili signali takta, odnosno dozvole rada, za sve ostale U/I portove. Ako je sadržaj brave F (False), nijedan U/I port se ne može čitati niti se u njega može upisivati. Pre čitanja U/I porta, program mora da smesti T (True) u bravu, a po završetku U/I operacije ponovo smešta F u bravu.

U mikroracunarima specijalne namene koji izvršavaju potpuno prečišćene programe, brava nije potrebna, ali u sistemima za razvoj softvera upotreba brave drastično smanjuje negativne efekte memorijsko preslikanog U/I, dok, s druge strane, omogućava da se iskaže većina prednosti ove U/I tehnike.

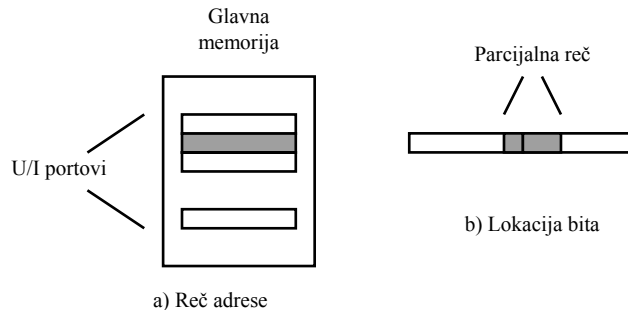
8.2. U/I portovi u adresnom prostoru

U/I portovi računara se moraju nalaziti u adresnom prostoru glavne memorije kod memorijsko preslikanog U/I, ili u odgovarajućem prostoru kod izolovanog U/I.

Adresa porta može biti kompletno ili nekompletno specificirana. Nekompletno specificirana adresa smanjuje hardver, ali sistem mora da vodi računa o tome da nikad ne koristi one adrese kojima nije nedvosmisleno definisano da li se pristupa memoriji, ili portu.

U/I portovi se obično smeštaju na jedno mesto u memoriji (slika 8.1a) a to dopušta da se ostatak memorije kontinualno koristi za RAM ili ROM, što olakšava posao kompilatora ili operativnog sistema. To takođe pojednostavljuje dizajn dekodera U/I portova, jer dozvoljava da nekoliko, ili svi U/I uređaji, dele veliki deo dekodera. U sistemima kod kojih je U/I uređaj na priključenoj ploči, a ta ploča ima RAM ili ROM, RAM, ROM i U/I portovi mogu biti grupisani zajedno u adresnom prostoru. Lokacija portova je netrivialan problem u sistemima

koji se masovno proizvode sa različitim kombinacijama uređaja u njima. Svaki pojedini uređaj treba da ima sopstvenu adresu bez obzira na koji sistem je priključen i koji uređaji su prisutni u sistemu. Alternativno, adresa porta treba da bude lako izmenljiva. Na primer, adresa se može nalaziti u adresnom registru mikroprocesora iz familije MC68000, a koristi se indeksno adresiranje za pristup portu. Najzad, relativna lokacija dva porta može pojednostaviti programiranje. Na primer, ako se dva porta, koji se uvek čitaju zajedno, nalaze u uzastopnim rečima, MOVEM instrukcija se može iskoristiti da se oba porta istovremeno pročita.



Sl. 8.1. Adresiranje U/I portova.

Ukoliko port ima dužinu manju od 16 bitova (1, 10 ili neku drugu) ostatak bitova biće nekorisna informacija koja treba da bude odbačena. Ako port nije levo poravnat u okviru reči, treba iskoristiti instrukciju logičkog pomeranja za njegovo poravnanje (slika 8.1b).

Na primer, ako je trobitni ulazni port na lokaciji \$4000 sa LS bitom na bit poziciji 6, tada će sledeća sekvenca instrukcija da izvrši njegovo čitanje i poravnanje u registru D0:

```

MOVE      $4000,D0; pročitaj ulazne podatke
LSR      #6,D0      ; poravnaj LS bit ulaznih podataka sa LS
                ; bitom registra D0
AND      #7,D0      ; odbacivanje viših bitova

```

Slično, prilikom operacije izlaza, podaci namenjeni portu moraju biti poravnani. Ako reč u memoriji mikroprocesora iz familije MC68000 ima više od jednog porta, podaci za druge portove se moraju iskombinovati sa podacima za port preko koga vršimo operaciju izlaza. Tako, ako podatke iz registra D0 šaljemo na port iz prethodnog primera, taj posao će obaviti sledeća programska sekvenca:

```

LSL      #6,D0      ;poravnaj LS bit registra D0 sa LS bitom porta
AND      dC0,D0     ;odbaci bitove koji ne idu u port
MOVE     $4000,D1;uzmi postojeće podatke iz ostalih portova
AND      #$FE3F,D1 ;ukloni bitove porta koji se menja
OR       D0,D1      ;kombinuj podatke u portu sa ostalim portovima
MOVE     D1,$4000;pošalji na izlaz sve bitove

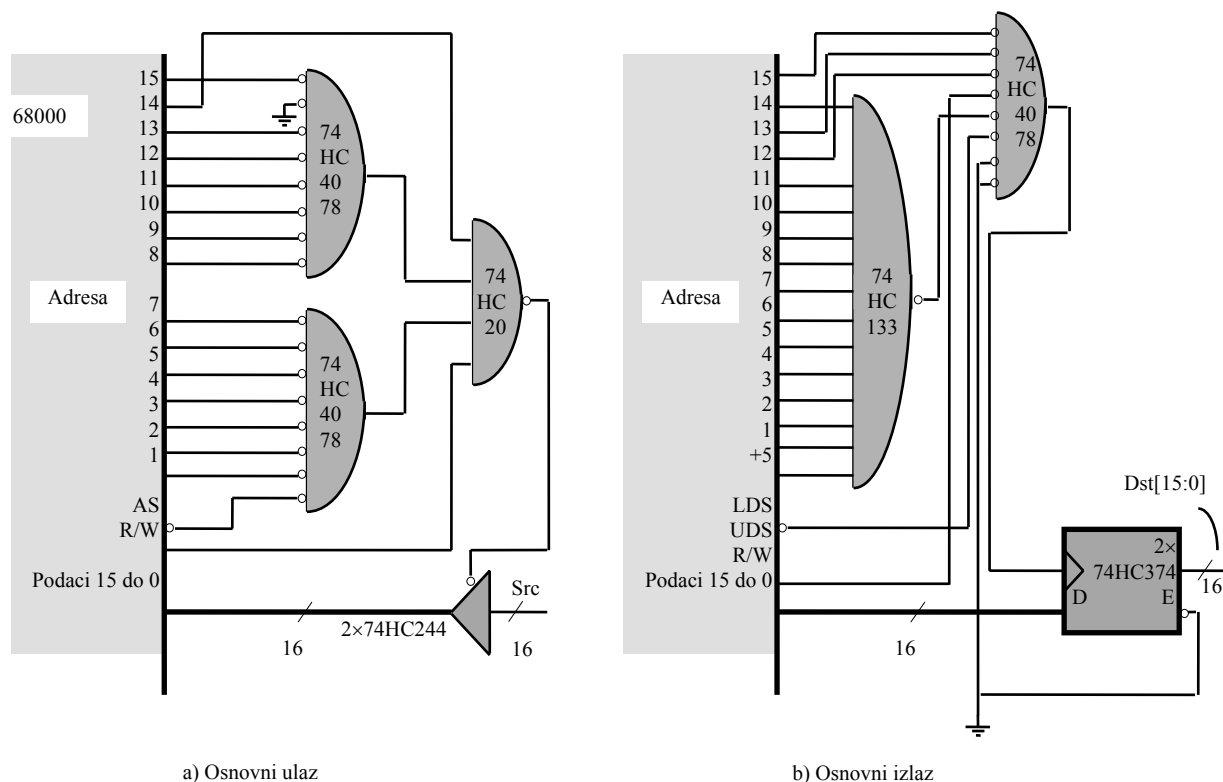
```

Naravno, prethodni programski segment se može pojednostaviti, ako je LS bit porta poravnat sa bitom 0 i nema drugih portova u okviru reči koja sadrži port. Tako se, da bi se pojednostavio softver, često stavlja samo po jedan port u reč, i to desno poravnat. Međutim, ovo ne mora bude najbolji način sa stanovišta hardvera.

8.3. Osnovni ulazni i izlazni portovi

Osnovni ulazni port je sposoban da uzorkuje signal kada mikroračunar izvršava ulaznu instrukciju (ili odgovarajuću instrukciju kod memorijsko preslikanog U/I) i da učita uzorak u registar podataka (ili da radi sa njim ako se on nalazio u nekoj memorijskoj lokaciji). Pošto većina mikroračunara koristi drajvere magistrale sa tri stanja, port mora da uzorkuje podatke na magistrali podataka upravo u trenutku kada mikroračunar izvršava komandu čitanja sa adrese dotičnog porta.

Tipični ulazni port prikazan je na slici 8.2a za slučaj operacije ulaza 16-bitnog podatka SRC (radi se o memorijsko preslikanom U/I) kad god se izvrši instrukcija MOVE \$4000,D0, ili njen ekvivalent. Signal čitanje/upis RW je na visoko kad god se vrši čitanje memorije ili U/I porta. Adresne i upravljačke su neodređenog stanja u prvom delu memorijskog ciklusa. Da bi se sprečilo da port pokrene magistralu podataka u konkurenciji sa ostalim drajverima, povlačeći veliku struju i generišući šum, ulazni port mora biti tako projektovan da nikada ne pokreće magistralu u tom trenutku. Adresni "strobe" signal AS je true (nisko) jedino kada prethodno pomenuti signali imaju određenu vrednost. Dekoder je potpuno specifikiran da vrši dekodiranje svih 15 adresnih bitova AŠ15 do 1C za adresu \$4000 i proverava da li je RW na visoko a AS na nisko.



Sl. 8.2. Logički dijagrami osnovnih U/I portova.

Izazni port treba obično da drži podatke na neodređeno vreme - sve dok ih program ne izmeni. Osnovni izlazni port je leč ili registar koji je sposoban da taktuje podatke sa magistrale podataka kad god mikroručunar izvršava izlaznu instrukciju (ili odgovarajuću instrukciju kod memorijsko preslikanog U/I). Magistrala podataka je priključen na ulazne priključke za podatke registra ili leča, a taktni signal je priključen na adresni dekodier tako da je registar taktovan kada mikroprocesor izvršava komandu upisa na adresi odabranoj za taj port.

Slika 8.2b prikazuje tipični osnovni izlazni port kod memorijsko preslikanog U/I, koji pamti i šalje na izlaz 16-bitnu reč podataka zapisanu na memorijskoj lokaciji \$4FFE. Podaci su neprekidno raspoloživi u vidu signala DST prema nekom spoljašnjem hardveru jer se signal dozvole E drži na nisko.

8.4. Specifičnosti U/I instrukcija mikroprocesora MC68020

MC68020 ima neke specijalne instrukcije namenjene U/I aktivnostima. To je pre svege bajt memorijska instrukcija, npr. MOVE.B D0,\$1000 i specijalna instrukcija MOVEP. Uz ove, postoje i instrukcije za rad sa bit poljima.

Instrukcija MOVE.B može da vrši transfer 8-bitnog podataka, što je naročito pogodno ako imamo 8-bitne portove. Instrukcija koja je projektovana za rad sa 8-bitnim podacima na 16-bitnoj magistrali podataka (ima je i MC68000) je MOVEP (Move Peripheral Data). Prenos podataka vrši se između specificiranog registra za podatke i bajtova memorije, počev od bajta na specificiranoj adresi. Najviši bajt registra za podatke smešta se prvi, a najniži bajt poslednji. Memorijska adresa specificira se korišćenjem adresnog registra i 16-bitnog razmeštaja. Ako je adresa parna, svi prenosi biće izvršeni na višoj polovini magistrale za podatke, a ako je adresa neparna korišće se niža polovina magistrale za podatke.

MC68020 ima i instrukcije za rad sa bitskim poljima, kao što su BFCLR, BFSET, BFINS, BFEXT itd. Ove instrukcije mogu da pristupe svakom pojedinom U/I portu dužine do 32 bita i da pri tom brišu, postavljaju, ubacuju, ili izdvajaju, respektivno, bit vektor proizvoljne veličine od 1 do 32 bita, nezavisno od poravnanja na granice bajta, reči ili duge reči.

8.5. Programabilni paralelni U/I uređaji

8.5.1. Problemi sa LSI U/I

Prvi mikroračunari vili su prilično razočaravajući, naročito u smislu ulaza/izlaza. Mada je CPU realizovan u LSI tehnologiji, U/I sekcija bila je implementirana desetinama SSI i MSI integrisanih kola. Ovi prvi mikroračunari nisu bilo po fizičkim dimenzijama mnogo manji od miniračunara jer je previše elektronskih kola bilo potrebno za U/I, a bili su i nešto sporiji od njih. Motivisani potrebom za kompaktnim U/I, proizvođači integrisanih kola počinju da razvijaju LSI integrisanih kola za U/I.

Prva opšta klasa U/I uređaja koja je izvedena u LSI tehnologiji bili su 8-bitni paralelni U/I uređaji. Serijski U/I uređaji i brojači su ovu promenu tehnologije doživeli nešto kasnije.

Dva ključna problema kod primene LSI kola je obim proizvodnje i broj pinova. Cena projektovanja takvih čipova je oko milion dolara, ali se takvi čipovi prodaju za oko deset dolara. Prema tome, oni se mogu projektovati samo ako se mogu prodati u količinama od više stotina hiljada komada. Zbog toga su ovakvi čipovi projektovani tako da se mogu upotrebiti u više sličnih aplikacija. Radi daljeg širenja upotrebe čipa, dodati su pinovi koji definišu koju funkciju obavlja taj uređaj. Ovi parametarski pinovi su obično priključeni na pozitivno napajanje ili masu. INTEL 8212 je odličan primer čipa sa parametarskim pinovima; pin načina rada MD priključuje se na nisko kada čip treba da radi kao ulazni uređaj, a na visoko kada treba da radi kao izlazni uređaj. Veći broj pinova povećava dimenzije čipa što znači da će čip zauzimati nešto više prostora na štampanoj ploči. Samo povećanje fizičkih dimenzija čipa unekoliko dioprinosi povećanju cene, ali je značajnije to da prostor na štampanoj ploči ima svoju cenu. Naime, ploča površine jedne kvadratne stope ima cenu i po nekoliko stotina dolara za голу ploču (bez instaliranih čipova).

Alternativni način upotrebe parametarskih pinova je da se ovi priključci stave unutar samog čipa, sa parametrima koji se pamte u upravljačkom registru ili upravljačkom portu. Ovakvi portovi su slični portovima za podatke u U/I čipovima u tom smislu da se njihov sadržaj može definisati odgovarajućom izlaznom naredbom. Ali, u ovom slučaju vrednosti koja su zapamćene u njima postavljaju parametre koji određuju funkciju čipa. Na primer, jedan bit u upravljačkom portu može da određuje kada će čip da radi kao ulazni a kada kao izlazni port. Ova tehnika rešava oba problema, obim proizvodnje i broj pinova; dopušteno je da se čip koristi u različitim sličnim aplikacijama, ali se pri tom ne zahteva veliki broj pinova za izbor određene funkcije čipa.

Manje bitan, ali sličan problem, je što neki LSI čipovi imaju previše portova a nedovoljan broj pinova. Adresa kojom se selektuje port se ponekad daje u delovima od po nekoliko bitova koji se, da bi se izbeglo korišćenje drugih pinova, smeštaju u druge portove unutar čipa. Efekat ovoga je da se nekoliko portova nalazi na istoj lokaciji i čitaju se instrukcijom kao što je MOVE \$8000,D0, na primer. Koji će port zaista biti pročitao zavisi od nekih bitova u drugom portu, tako da se mora smestiti odgovarajuća reč u taj drugi port da bi mogli da pročitamo port koji želimo.

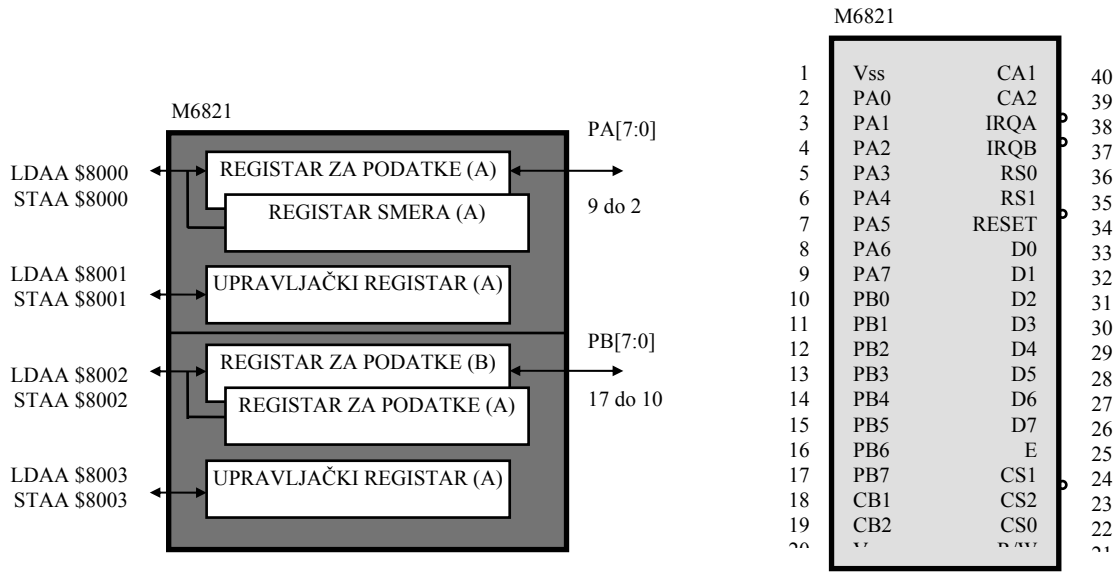
Šta više, čini se da svako rešenje donosi i novi problem. Upravljački port mora biti inicijaliziran da bi se postavili parametri kad se mikroračunar uključi. Ovo se mora obaviti pre nego što se uređaj počne koristiti ili neposredno posle uključenja napajanja. Inicijalizacija upravljačkog porta konfiguriše uređaj. Inicijalizaciona rutina često liči na nekakav mračni ritual koji je teško objasniti i razumeti ali se relativno lako izvodi. Programer mora da odredi koji se bitovi u upravljačkoj reči moraju postaviti da bi se čip osposobio da vrši određenu funkciju, kao i korektnu sekvencu kojom se ti bitovi postavljaju. Treba istaći da je ova tehnika naročito sklona softverskim bagovima.

8.5.2. MC6821 periferni interfejs adapter (PIA)

MC6821 je LSI intergrirano kolo projektovano za paralelni U/I koje sadrži i upravljački port radi povećanja fleksibilnosti čipa. Ovaj čip se takođe zove i periferni interfejs adapter (Peripheral Interface Adapter) ili PIA, ali ovo nije jedini čip iz Motoroline familije koji nosi oznaku PIA.

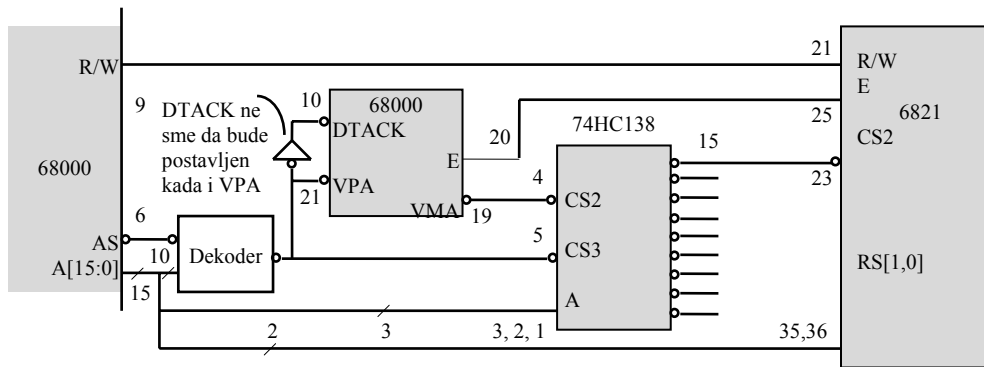
Čip sadrži u sebi dva gotovo identična U/I uređaja; svaki uređaj ima port podataka koji može biti ili ulazni ili izlazni port, ili delom ulazni a delom izlazni port. Svaki uređaj ima osam pinova (pinovi periferalnih podataka), koji prihvataju podatke iz spoljašnjeg sveta kada uređaj radi kao ulazni port, ili predaju podatke spoljašnjem svetu kada uređaj radi kao izlazni port. Svaki uređaj u MC6821 ima tri 8-bitna porta: port podataka, direkcioni i upravljački port (slika 8.3). Port podataka može biti ulazni ili čitljivi izlazni port. Direkcioni port određuje, na bit po bit osnovi, kada se bit porta podataka tretira kao ulazni bit a kada kao čitljivi izlazni bit. U većini uobičajenih slučajeva, svi direkcioni bitovi su na T ili na F, što čini ceo port ili 8-bitnim čitljivim izlaznim portom ili 8-bitnim ulaznim portom, respektivno. Trima portovima u okviru jednog uređaja može se pristupiti samo pomoću dve reči memorije na taj način što jedan od bitova upravljačke reči predstavlja dodatni adresni bit koji selektuje ili port podataka, ili direkcioni port. Na primer, ove dve reči mogu biti na lokacijama \$8000 i \$8001. Ako program čita ili upisuje na \$8001, on uvek čita iz, ili upisuje podatke u upravljački port. Ako program čita ili upisuje u lokaciju sa

adresom \$8000, on pristupa direkcionom portu ako je bit 2 upravljačkog port na F, ili pristupa portu podataka ako je taj bit bio na T.

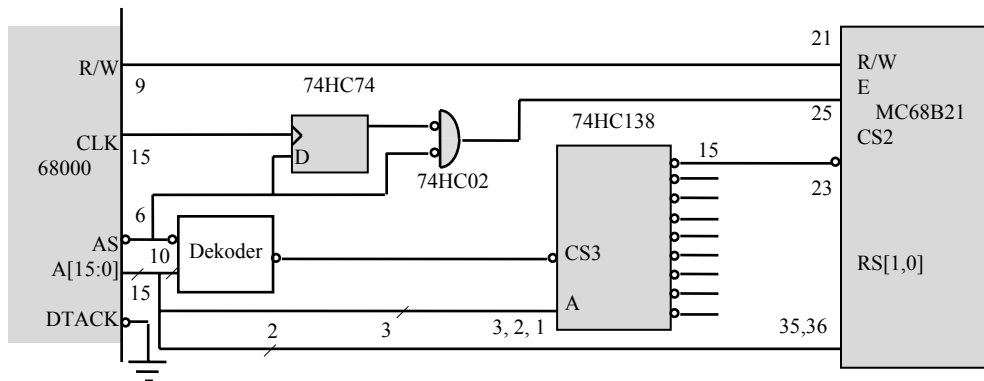


a) Pojednostavljeni blok dijagram

b) Povezivanje pinova



c) Sinhronizacija periferala 6800 korišćenjem VPA



d) Sinhronizacija periferala korišćenjem E Clock kola

Sl. 8.3. MC6821.

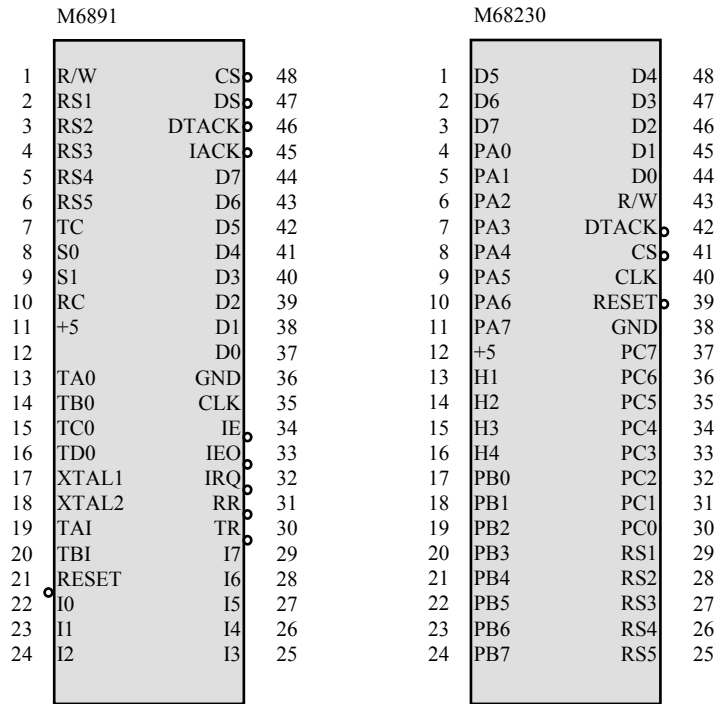
Oznake pinova za MC6821 prikazane su na slici 8.3b. Ovaj čip može se povezati na MC68000 na dva načina. MC68000 je bio projektovan pre nego što su za njega postali raspoloživi specijalni U/I čipovi, tako da je predviđen specijalni "memorijski ciklus" za pristup postojećim 8-bitnim MC6800 U/I čipovima. Signali VPA (Valid Peripheral Address), VMA (Valid Memory Address) i E takt koriste ovu tehniku i mogu se priključiti kao na slici 8.3c. Kolo koje generiše E signal je jednostavno za implementaciju i radi sa mikroracunarima zasnovanim na MC68000 i MC68020, a prikazano je na slici 8.3d.

8.5.3. MC68230 i MC68901

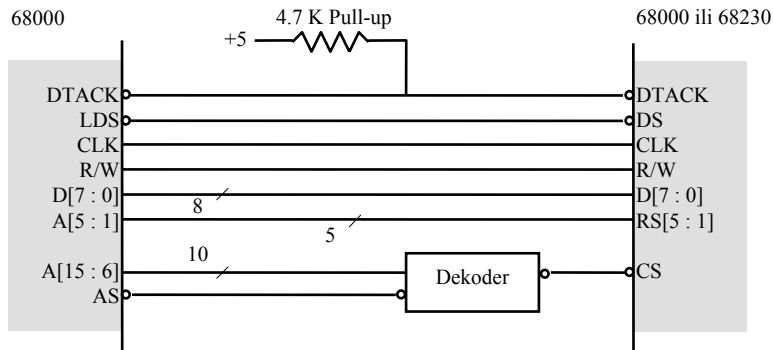
Višefunkcijski periferal (Multi-function Peripheral - MFP) MC68901 i Paralelni interfejs/tajmer (Paraller Interface/Timer - PI/T) MC68230 su U/I čipovi projektovani za spregu direktno sa MC68000, a obuhvataju jedan ili tri paralelna U/I uređaja slična onim u MC6821, među ostalim funkcijama.

Oznake pinova za MC68901 i MC68230 prikazane su na slici 8.4a i 8.4b, a način povezivanja na MC68000 dat je na slici 8.4c.

MC68901 obuhvata četiri tajmera i serijski komunikacioni uređaj (o kojima će biti reči kasnije), kao i paralelni uređaj o kome ćemo sada govoriti. Blok dijagram MC68901, dat na slici 8.5a, prikazuje samo paralelni uređaj. Pretpostavimo da se čip nalazi na adresama od \$8000 do \$802F. Svaki bit u direkcionom portu, na adresi \$8004, određuje da li će odgovarajući bit u portu podataka na adresi \$8000, biti ulazni ili izlazni. Ovo je potpuno isto kao i kod MC6821 osim što se portovi ne nalaze na istim lokacijama, i što nije potreban upravljački port da razdvoji ova dva porta.



a) Povezivanje pinova kod M6891 b) Povezivanje pinova kod MC68230



c) Povezivanje izmeju 68000 i 68230 ili 68901

Sl. 8.4. MC68000 kompatibilni uređaji sa paralelnim U/I.

MC68230 obuhvata tajmer o kome će kasnije biti reči, kao i tri paralelna uređaja o kojima ćemo sada govoriti. Blok dijagram ovog čipa je na slici 8.5b. Uređaji su obeleženi sa A, B, i C; uređaji A i B su identični, uređaj C je jednostavniji. Uređaj C ima svoj direkciono port i ponaša se na identičan način kao i paralelni uređaj u MC68901. Da bi svi bitovi porta podataka bili ulazni, potrebno je obrisati lokaciju \$8008, Da bi svi bitovi porta bili izlazni, treba u pomenutu lokaciju smestiti vrednost \$FF. Međutim, uređaj C deli svoje pinove sa specijalnom logikom čipa MC68230 koja se koristi za prekide i DMA. Ako se koriste ove specijalne funkcije, pinovi nisu raspoloživi za opšti U/I.

Reći ćemo sada nekoliko reči o uređaju A, dok je uređaj B analogan. Uređaj A ima način rada u kome su sve njegove funkcije iste kao i funkcije uređaj C, i izvestan broj načina rada koji koriste tehniku dvostrukog baferovanja. Dvostruko baferovanje je tehnika koja služi za prenos informacija između računara i periferala, gde računar deluje na sadržaj jednog bafera, dok informacija iz drugog bafera predstavlja ulaznu ili izlaznu informaciju. Dvostruko baferovanje za slučaj izlaza je prikazano na slici 8.5c. Instrukcijom MOVE D0,\$8010 se podaci smeštaju u BUFFER1 odakle se, što je pre moguće, kopiraju u BUFFER2. Kako je na taj način BUFFER1 oslobođen, sledećom MOVE operacijom se mogu smestiti dodatni podaci u njega. Kada su oba bafera puna, postavlja se bit u statusnom registru. Kada je barem BUFFER2 pun, signal na pinu H2 obaveštava spoljašnju logiku da su podaci raspoloživi. Kada spoljašnja logika koristi podatke, ona šalje ivicu signala na pin H1, čime su javlja uređaju da se podaci koriste. Kada se ivica signala javi na H1, tada, ako se podaci nalaze u BUFFER1, oni se prenose u BUFFER2, a bit u statusnom registru se briše jer je oslobođen prostor za nove podatke. Ovaj statusni bit se može softverski ispitivati da bi se odredilo kada je moguće, a kada nije, smeštanje novih podataka u BUFFER1.

Ova tehnika se zove *dvostruko baferovanje*, jer se koriste dva baferska registra, a opisan je jedan od mogućih načina rada za ovu tehniku. Kod drugog načina rada, statusni bit ukazuje da su oba registra prazna, a ne samo jedan, i na taj način se program obaveštava da su svi podaci poslani.

Dvostruko baferovanje za slučaj izlaza je prikazano na slici 8.5d. Ivica signala na pinu H1 (na primer, prednja ivica signala) izaziva smeštanje podataka prisutnih na pinovima za podatke, u baferski registar BUFFER. bit statusnog registra se može testirati softverski da bi se utvrdilo da li su podaci raspoloživi u baferu a pin H2 signalizira spoljašnjem hardveru da nema više slobodnog prostora. Instrukcija MOVE \$8010,D0 čita sadržaj bafera BUFFER a pin H2 sada ukazuje da se novi podaci mogu smestiti u bafer. Međutim, instrukcija MOVE \$8014,D0 čita podatke direktno iz ulaznog porta. Čak iako se samo jedan registar koristi za ulaz, ova tehnika se takođe zove dvostruko baferovanje, jer je simetrična dvostruko baferovanom izlazu.

Osnovna prednost dvostrukog baferovanja je u tome što procesor može da prenosi podatke bez precizne sinhronizacije sa spoljnim svetom. Na ulazu, spoljni svet puni registar BUFFER i to signalizira uređaju, a procesor može da čita te podatke kasnije, u bilo kom trenutku, sve dok novi podaci ne pristignu na ulaz. Slično, na izlazu, dvostruko baferovanje dozvoljava procesoru da postavi nove podatke čak i dok se stari podaci nalaze na izlazu a nije započelo njihovo korišćenje. Dvostruko baferovanje omogućava prenos podataka između sistema sa različitom sinhronizacijom, i to najvećom mogućom brzinom.

Da bi uređaja A radio kao jednostavan čitljivi izlazni uređaj, potrebno je obrisati MAIN MODE registar, smestiti \$80 u upravljački registar uređaja A, i smestiti jedinice u odgovarajuće bitove direkcionog registra. Pretpostavljajući se MC68230 adresira počev od lokacije \$8000, i da svi bitovi porta treba da budu izlazni bitovi, odgovarajuću inicijalizaciju će izvršiti sledeći programski segment:

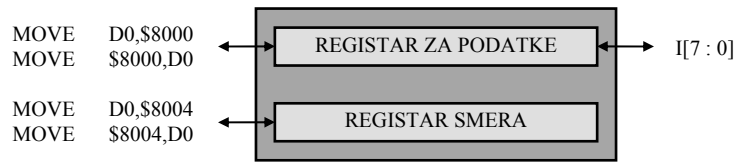
```
CLR      $8000
MOVE    #0,$800C
MOVE    #$FF,$8004
```

Da bi načinili uređaj A dvostruko baferovanim izlaznim uređajem, potrebno je obrisati MAIN MODE port, smestiti \$70 u CONTROL port uređaja A, i smestiti jedinice u odgovarajuće bitove direkcionog porta kako bi se naznačilo koji su bitovi porta podataka izlazni, i najзад, postaviti bit 4 u portu MAIN MODE. Svi bitovi će postati izlazni ukoliko se izvrši sledeća sekvenca:

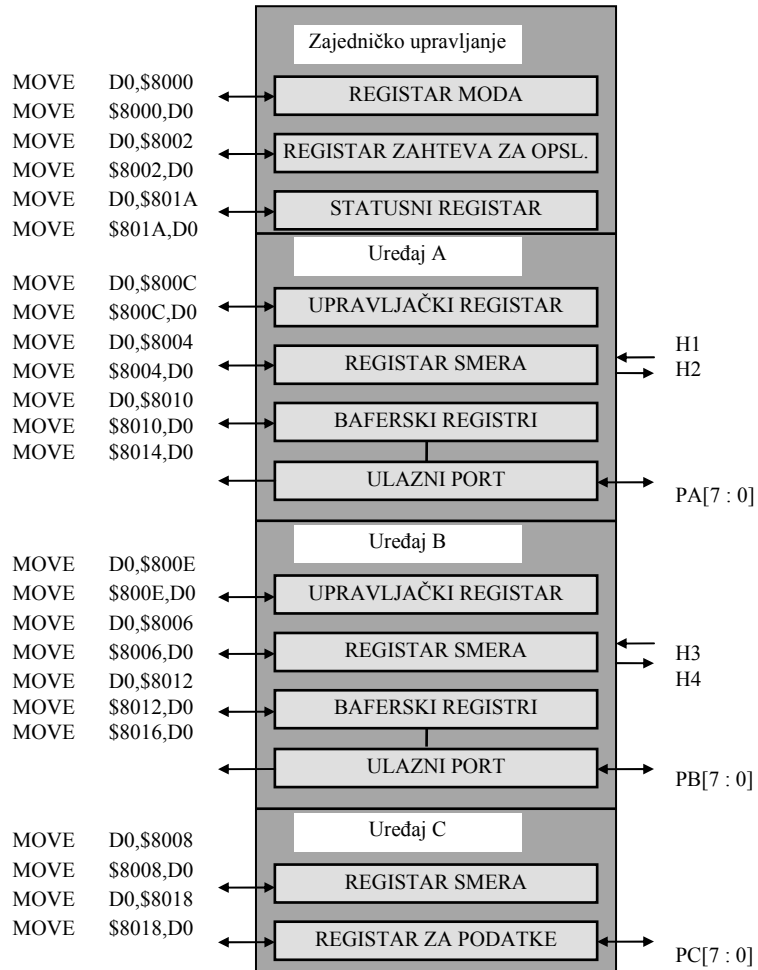
```
CLR      $8000
MOVE    #0,$800C
MOVE    #$FF,$8004
MOVE    #0,$8000
```

Da bi uređaj A radio kao dvostruko baferovani ulazni uređaj, potrebno je obrisati MAIN MODE port, smestiti \$60 u port CONTROL uređaja A, smestiti nule u odgovarajuće bitove direkcionog porta, i postaviti bit 4 u MAIN MODE portu. Svi bitovi će biti ulazni posle izvršenja sekvence:

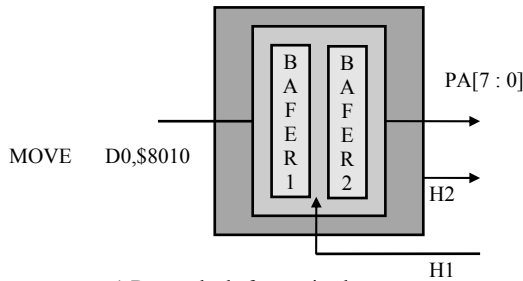
```
CLR      $8000
MOVE    #0,$800C
CLR      $8004
MOVE    #0,$8000
```



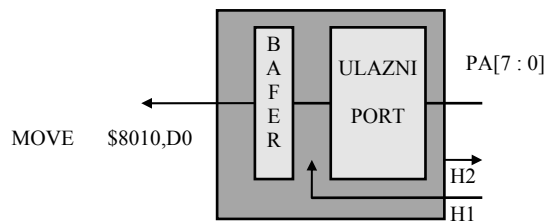
a) Blok dijagram M68901



b) Blok dijagram M68230



c) Dvostruko baferovanja ulaza



d) Dvostruko baferovanja izlaza

Sl. 8.5. Blok dijagrami čipova MC68901 i MC68230.

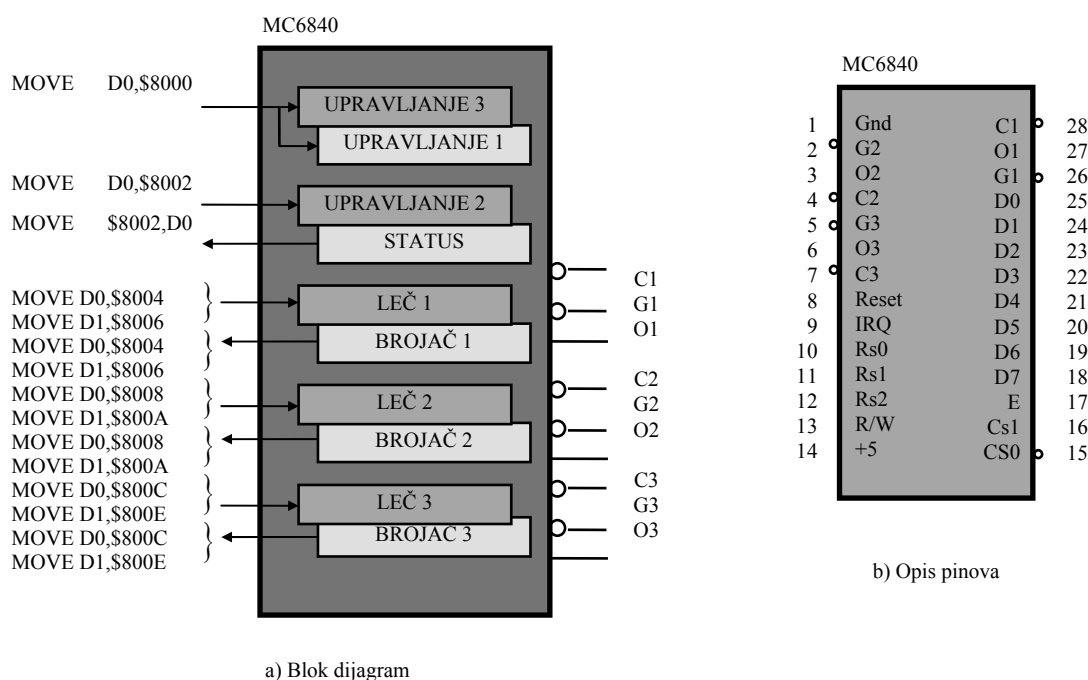
8.6. Brojači i tajmeri

Brojač/tajmer je jedan od najfleksibilnijih čipova koji se može staviti u mikroracunar. Ovaj čip generiše kvadratni talasni oblik, koji se može iskoristiti za dobijanje sinusnog, ili bilo kog drugog periodičnog talasnog oblika. Ovakvi talasni oblici imaju, dalje, veliku primenu u različitim uređajima.

8.6.1. MC6840 brojač/tajmer

Brojački/tajmerski sistem raspoloživ je kao integrisano kolo, kao MC6840. Ovaj čip je projektovan za MC6800 familiju osmобitnih mikroprocesora i može se koristiti sa svakim računarem.

Čip MC6840 ima tri nezavisna brojačka/tajmerska uređaja. Svaki uređaj ima šesnaestobitni samočitajući brojački registar, odgovarajući šesnaestobitni samoupisni leč radi reinicijalizacije brojača svaki put kada brojač dostigne nulu, i osmобitni samoupisni upravljački registar. Tri brojača imaju zajednički samočitajući statusni registar radi identifikovanja prekida. Svaki uređaj ima tri priključka na spoljni svet: takt C sa negativnom logikom, gejt G sa negativnom logikom, i izlaz O sa pozitivnom logikom. Slika 8.6a prikazuje blok dijagram modula, a slika 8.6b prikazuje raspored pinova.



Sl. 8.6. MC6840.

MC6840 ima devet pinova za vezu sa spoljnim svetom (po tri za svaki uređaj), i devetnaest pinova za vezu sa MC68000 (vidi slika 8.6b). Osm pinova za podatke priključeni su na odgovarajuće linije magistrale podataka. Napajanje, uzemljenje, linija za resetovanje magistrale, i RW linija priključeni su na odgovarajuće pinove. Pinovi za selekciju registara RS0, RS1 i RS2 su obično priključeni na adresne linije jedan, dva i tri, respektivno. Takođe postoji i izvestan broj pinova za upravljačke signale o kojima nećemo govoriti.

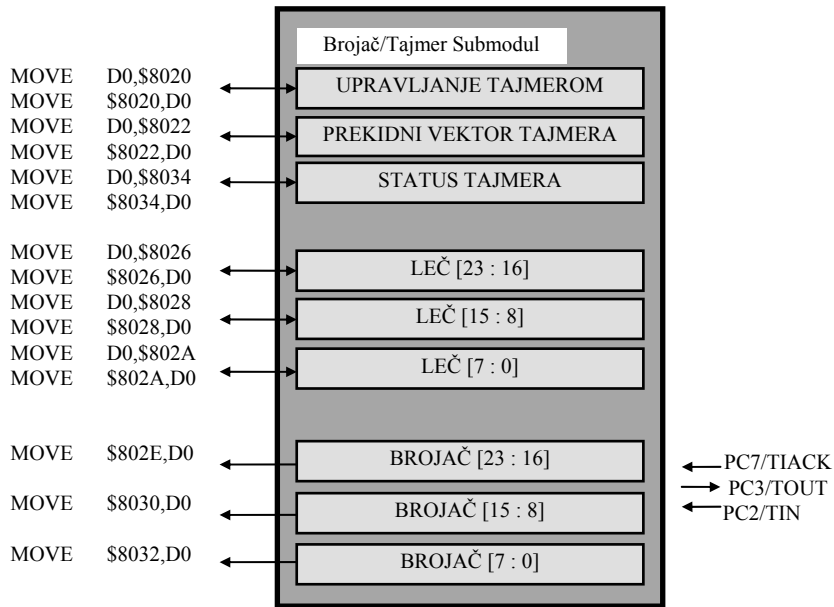
Pretpostavimo da su adrese čipa od \$8000 do \$800F, pri čemu se zauzimaju niži bajtovi svake reči. Reči upravljačkog porta i statusnog porta su na lokacijama \$8000 i \$8002, a svaki leč-brojački port se nalazi na po jednom paru lokacija počev od parnih adresa.

8.6.2. Pregled brojača/tajmera u čipovima MC68230 i MC68901

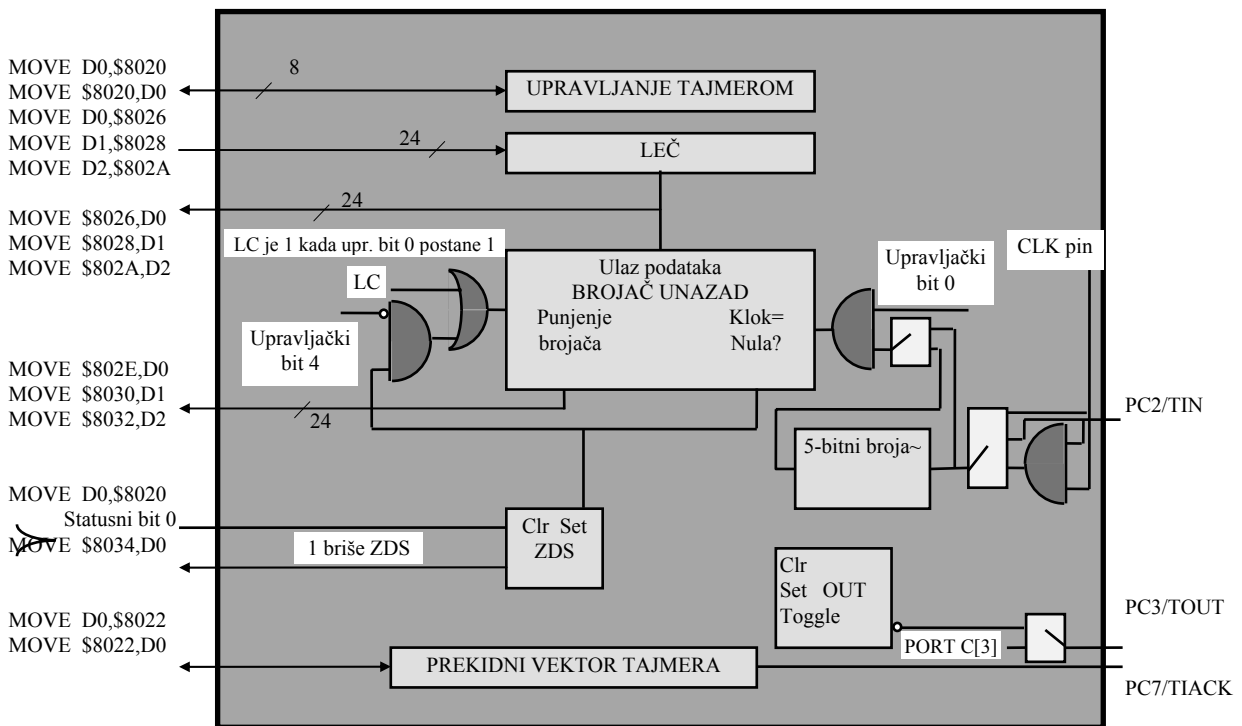
Brojačko/tajmerski podsistem čipa MC68230 PIA prikazan je na slici 8.7a. Dvadesetčetvorobitni registar COUNTER se dekrementira, a pošto se nađe u stanju nula, puni se iz 24-bitnog registra LATCH. Tajmer ima upravljački port CONTROL i statusni port STATUS, slično MC6840. Uz to, čip poseduje i port TIMER VECTOR INTERRUPT koji šalje broja vektora prekida kada se za ovaj brojač/tajmer koriste spoljni vektorski prekidi.

MC68230 ima tri pina koji se mogu koristiti kao i pinovi uređaja C, ili kao pinovi brojača/tajmera. Bit sedam uređaja C se može alternativno koristiti sa brojačem/tajmerom kao potvrda prekida (TIACK) kod tajmerskih prekida za sporazumevanje sa MC68000. Bit dva uređaja C se alternativno može koristiti kao ulaz (TIN) u brojač,

gde će se brojati impulsi koji se sa njega primaju. Bit tri uređaja C može biti korišćen i kao izlaz (TOUT) iz brojača/tajmera, bilo za generisanje kvadratnog talasnog oblika, bilo za generisanje signala zahteva za prekid.



a) Mapa registara



b) Blok dijagram

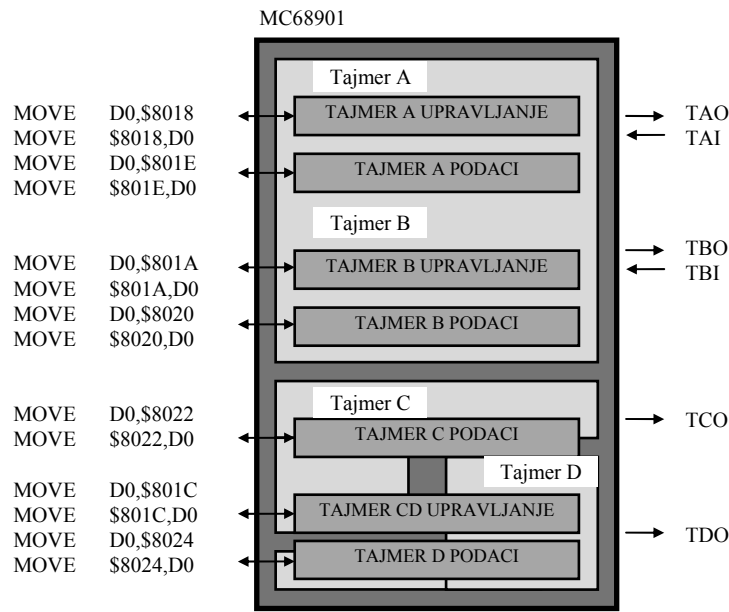
Sl. 8.7. MC68230 brojački/tajmerski podsistem.

Slika 8.7b. prikazuje blok dijagram čipa MC68230. Ovaj čip je konceptualno sličan čipu MC6840. Mehanizam koji taktuje brojač obuhvata petobitni delitelj frekvencije, tako da se brojač može dekrementirati frekvencijom taktnog signala podjeljenom sa 32. Dekrementiranje se javlja kada upravljački bit 0 ima vrednost logičke jedinice. Brojač i leč se nalaze u uzastopnim lokacijama, ali ako se brojač dekrementira za vreme čitanja/upisa u/iz njega, dobijeni podaci mogu biti nekorektni. Da bi se osigurali korektno čitanje i upis u brojački registar, treba obrisati upravljački bit nula kako bi se zaustavilo brojanje. Podatak n u registru LATCH se prenosi u DOWN COUNTER kada upravljački bit postane jednak 1, ili ako je upravljački bit 4 jednak 0 a brojač se nalazi u stanju nula. Ako je upravljački bit 4 jednak 1, DOWN COUNTER broji po modulu 2, a inače po modulu n-1. Flip-flop detekcije nultog stanja (ZDS), čita se kao LS bit lokacije \$8034, postavlja se kada brojač dođe do nule, a briše se upisom jedinice u LS bit reči na adresi \$8034.

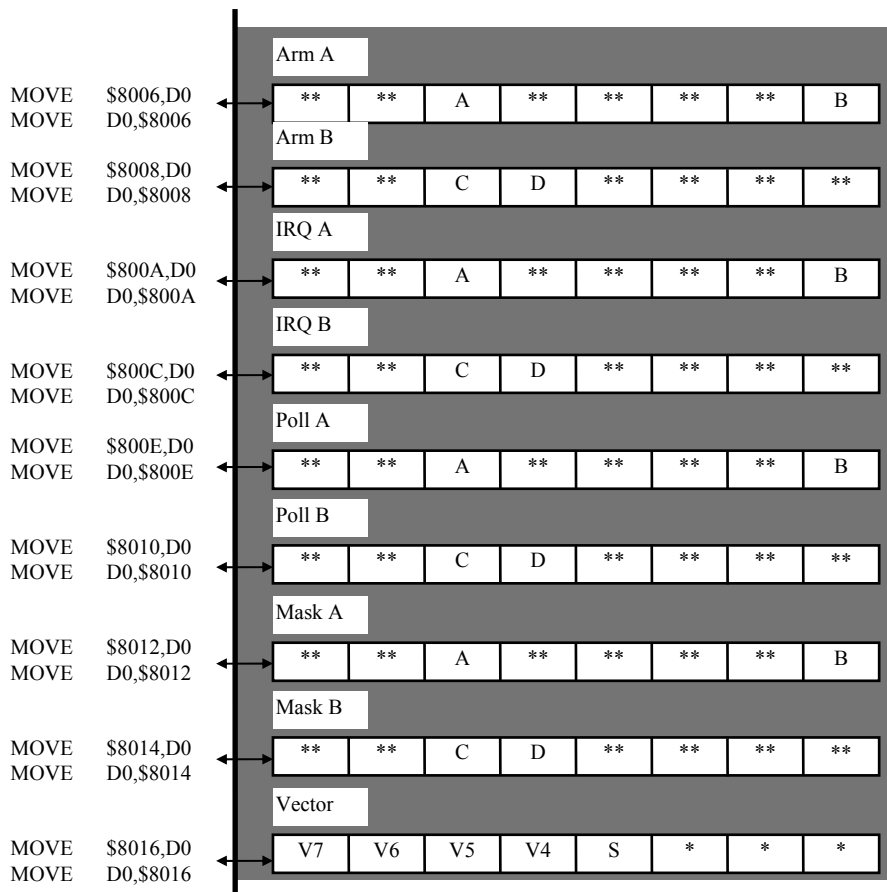
MC68901 MFP je MC68000 kompatibilni U/I čip. Njegov brojački/tajmerski podsistem prikazan je na slici 8.8a i 8b. Koriste se četiri modula, prva dva (A i B) su identična a druga dva (C i D) su unekoliko redukovana verzija brojača/tajmera A. Navešćemo nekoliko detalja o brojaču/tajmeru A.

Kao što se vidi sa slike 8.8a, brojač/tajmer A ima upravljački port TIMER A CONTROL i port podataka TIMER A DATA. Port podataka je u stvari 8-bitni brojač unazad i leč. Upisom podataka u odgovarajuću lokaciju izvršiće se upis u leč, koji ih takođe može upisati u brojač. čitanjem lokacije čita se brojač. Kao što je prikazano na slici 8.8b, brojač/tajmer A takođe ima bitove porta koji su deljivi sa paralelnim U/I portovima. Kod portova ArmA, IRQA, PollA i MaskA, bit 5 predstavlja statusne i upravljačke bitove za brojač/tajmer A.

Slika 8.9. prikazuje organizaciju brojača/tajmera A. Osmobitni COUNTER se može taktovati oscilatorom ili sa TAI ulaza. Oscilator može biti kontrolisan kristalom, ili se na XTAL1 može dovoditi kvadratni talasni oblik. Njegovi izlazi se dele brojem između 4 i 200.



a) Specijalni portovi tajmera/brojača



* Nekorišćeni bitovi ** Bitovi koje koriste paralelni portovi ili UART delovi čipa

b) Portovi deljivi sa drugim podsistemima

Sl. 8.8. MC68091 brojački/tajmerski podsistem.

