

Ako može neko da mi kaže koja vrednost se nalazi u registru AX (ostali podaci su DS=0004h , BX=3h).

MOV AX, [BX] ; dakle u AX ide vrednost sa adrese DS:BX a posto je u postavci zadatka receno da je:

"Sadržaj memorijske lokacije (bajta) jednak vrednosti nizeg bajta adrese te lokacije" onda bi u AX trebalo da ide vrednost nizeg bajta adrese 0004:0003 ?

I to je samo jedan bajt a šta ćemo sa drugim posto je AX veličine 2 bajta?

Formirao adresu:

$DS \cdot 10 + BX = 00043$

Sadržaj adrese je najniži bajt te adrese, za 00043 je 43, za 00044 je 44

Posto se radi sa recima, citaju se dva bajta, sa lokacija 00043 i 00044, tj. citaju se vrednosti 43 i 44.

Ako se dobro secam, visi bajt reci se nalazi na visoj, a nizi na nizjoj adresi, pa je trazena rec 4443

=====

add AX, [BX]

$AX += MEM[DS:BX] = MEM[DS \cdot 10H + BX] = MEM[4 \cdot 10 + 28] = MEM[68H]$

posto se adresira bajt, mem[68]=68h po uslovima zadatka a mem[69h]=69h

AX je 2 bajta pa imamo: AX=6966h, add AL, [BP][SI],

$al += mem[SS:(BP+SI)] = MEM[B \cdot 10 + 76+34] = MEM[15A]$ po uslovima je Mem[15AH]=15H

pa je al=68+5a=C2, AX=69C2, add AX, [BX+4][SI],

$AX += MEM[DS:(BX+8+SI)] = MEM[40+28+4+34] = MEM[A0]$, pa je AX=69C2+ A1A0=10B62=0B62

e sad za poslednje ne dobijam isto pa ako neko vidi gresku neka vice.

Neka pitanja sa ponudjenim odgovorima koja su se javljala ranije:

metode poboljšanja ILP

-treba da se pise sve, ne samo one 3 tehnike prevazilazenja data dependency, iako je to dato u nekom starom roku kao odgovor. :S

baud rate factor

-u debeloj knjizi - na 489str

da li postoji situacija kada nije moguc AEOI mod rada

-ukoliko omogucimo AEOI onda bi se desila situacija da po prihvataju prekida od strane procesora, drugi prekid manjen prioriteta bude prihvacen, a ovaj sto je bio viseg prioriteta bude odbacen. VALJDA

Da li ima smisla planirati da se dalja povecanja broja tranzistora utroše na za povecanje broja paralelnih tokova u SS (SuperScalar) procesoru kako bi se povecala brzina procesora? Precizno i kratko obrazložiti odgovor.

-na ova pitanja je uvek cupavo odgovarati jer je ona Veljkova knjiga bajata i neke ideje su itekako zastarele . Ne zato sto nemoze da se izvuče više paralelizma iz koda. Prave zavisnosti po podacima , i delimicno kontrolne zavisnosti limitiraju brzinu izrsavanja koda. Pravljenje dubljeg i sireg pipeline-a nema više smisla jer siri ne može više da se popuni instrukcijama a dublji će imati još više stall-ova sto će dovesti u stvari do usporenja procesora. Tu svakako treba misliti na potrošnju procesora koja je danas velika i koja bi se ne racionalnim trošenjem tranzistora još povećala.

Na assembleru 8086 napisati rekurzivnu funkciju koja racuna sumu prvih N elemenata.

-treba da stavljas argumente preko steka i da po povratku sa steka skines sve argumente koje si stavio... Na pocetku recimo stavis N udjes u funkciju i onda smanjujes N dodajes na sumu i proveravas kada ce postati 0 da krenes da se vracas...

Kada podaci treba da budu spremni na magistrali mikroprocesora 8086 u ciklusu citanja? Šta se radi da to uvek bude zadovoljeno?

-Treba da budu spremni do pocetka trece faze. Da bi se osigurao da ce podaci sporih periferija zaista biti spreni umeće se wait stanje onoliko taktova koliko je to potrebno

Navesti dva primera kako se štedi memorija u mikrokontroleru 8051. Na kom nivou i ko pravi ove uštede?

-Pa prvo on ima integrisanu memoriju u sebi, drugo te memorijske banke koje spasavaju stek i treće neke lokacije su bit-adresabilne

Zašto se kod 8086 periferijama ne pristupa pomocu instrukcije MOV? Da li bi se ova instrukcija mogla upotrebiti za pristup periferijama? Ako bi – kako, ako ne – zašto?

-Zato sto ima razdvojen memorijski sistem , periferijama se pristupa sa in i out. Mogla bi da se koristi kada bi memorijski prostor bio u preslikan

Da li su sva 4 kanala DMA kontrolera identicna u pogledu operacija koje mogu da izvedu? Ako jesu, navesti sve operacije koje su moguće, ako nisu, navesti jednu operaciju po kojoj se neki kanal razlikuje od ostalih.

-Kanal 0 može da zadrži istu adresu za sve transfere i time se omogućuje da se jedna rec kopira u ceo blok memorije.

Problem izgladnjivanja kod kontrolera prekida...

-Moguće je. Koristi se deisy chain rezim rada. Tada se posle svakog primljenog prekida obara prioritet istog.

8251 i greške pri prijemu niza bitova

-pretpostavljam da je to iz prezentacije Vježbe-radna verzija.ppt, slika na strani 164/225:

prvi bit u nizu koji je nula- to je start bit

onda idu data bitovi(broj bitova je zadat u parametrima prenosa)

zatim bit parnosti - kako se racuna parnost ?(tj. u kom slucaju je bit parnosti 1- kada je broj jedinica u data delu paran, ili kada je neparan; u zavisnosti parna/neparna parnost)

zatim ide stop bit(jedan ili više?) cija je vrednost 1

Moguće je detektovati greške (strana 163/225):

Ako je stop bit 0, frame error,

Ako parnost nije OK, parity error,

Ako se primi jos jedan znak prije citanja prethodno primljenog, OVERRUN error.

Evo kako ja mislim da se radi onaj sa serijskim portom:

Imas znaci taj niz bita koji od prilike izgleda ovako: 11111101111101000111111

I kaze da se prenosi 6 bita kao poruka, ima bit parnosti i kaze da citanje iz bafera jos nije pocelo a ova poruka stize.

E sad prva nula u nizu bitova oznacava start bit i posle te nule ide 6 bitova koji cine poruku i to su 111110. Posle ovih 6 bitova poruke ide bit parnosti i kaze se u tekstu da je parnost podesena na neparno sto mu valjda znaci da broj jedinica koje se posalju u poruci + bit parnosti treba da bude neparan broj jedinica. E sad posto se u poruci salje 5 jedinica a bit parnosti je isto 1, sto daje paran broj jedinica (6) znaci postoji ta greska parnosti.

Posle bita parnosti treba da ide valjda jedna ili dve jedinice koje oznacavaju kraj niza koji se salje, medjutim posle bita parnosti ovde ide par nula tako da se generise frame error.

U zadatku se kaze da ciklus citanja iz 8051 jos nije pocao kad stize nova poruka, pa se zato generise greska koja kaze da se u baferu prepisuju necitani podaci.

Eto ja mislim da se tako radi ovaj zadatak, nisam 100% siguran da mi je tacan, ali položio sam sad mips pa verovatno je jedan od dobrih.

===mozes da dodas na pocetak i kraj proizvoljan broj 1, tako sam ga barem razumeo.===

===Start bit je uvek obaranje vrednosti na 0, a stop bit je postavljanje na 1 na kraju cele poruke, e sad u mod word-u koji se zadaje 8051 najgornja 2 bita te reci oznacavaju koliko ce stop bita biti (1, 1 1/2 ili 2; nije mi samo jasno kako može da bude 1 ipo al tako pise u knjizi). U toj istoj reci za mod postavljas da li ce postojati bit parnosti i koliko se bita prenosi u poruci. Ako postoji bit parnosti, on pretstavlja prvi bit posle bita poruke a posle njega dolaze stop bit(ovi).

Ovo sve mozete procitati u knjizi "Microprocessors and Interfacing" od Douglas V. Hall u poglavlju 14. Inace fenomenalna knjiga za spremanje projekta, a mnogo dobro dodje i za sam ispit, verovatno najbolji udzbenik koji sam procitao na faxu!

====Podesavanje trajanja stop bita (1, 1.5 ili 2) se odnosi iskljucivo na slanje. Na prijemu 8251A uvek proverava da li postoji jedan stop bit. Ne smeta ako ima vise stop bita, tj. ako jedinica traje izmedu znakova.====

da li se u 8-bitnoj periferiji u mikroprocesorskom sistemu zasnovanom na 8086 koja zahteva vise adresa mogu dodeliti uzastopne adrese? Obratiti odgovor.

-Ne može jer ako je periferija 8-bitna mi na nju dovodimo recimo donjih 8 linija data magistrale i one su aktivne na svaku parnu adresu. Ako bi se istoj periferiji dodelila jos jedna адреса koja ce da bude prva sledeca, to ce biti neparna адреса, a neparna адреса aktivira gornjih 8 linija data magistrale koje nisu prikljucene na ovu periferiju.

Ucitati vrednost sa adrese PROM u registar SI i sa adrese PROM+2 u registar DS na najefikasniji nacin.Isto ponoviti za DI i ES koristeći lokacije PROM+4 i PROM+6

-Valjda ovako:

-LDS SI, PROM

-LES DI, PROM+4

Zasto se kod 8086 periferija na pristupa pomocu instrukcije MOV?Da li bi se ova instrukcija mogla upotrebljavati za pristup periferijama? Ako bi-kako ako ne -zasto ne?

-Nisam 100% siguran za ovo al mislim da bi moglo sa MOV da se pristupa i periferijama, tako sto napravimo dekodir koji jednostavno za odredjene adrese sa adresne magistrale aktivira periferije koje trebaju a memoriju drzi neaktivnom, problem kod

ovoga je sto adrese koje su namenjene periferijama u ovom slucaju nece biti dostupne u samoj memoriji, a s obzirom da ne koristimo IN i OUT instrukcije signal M/IO nam nije potreban.

Koja su dva moguca pristupa pri rješavanju resursnog konflikta u superskalarnom procesoru u kojem izvršavanje ima više faza? U jednoj recenici uporediti performanse i složenost ova dva pristupa.

-Register renaming i out-of-order issue

====Pretpostavljam da je ovo pitanje nastalo tako što je neko slučajno otvorio knjigu i na vrhu 50. strane uocio sliku sa komentarom (two alternative solutions - citaj dva moguca pristupa pri rješavanju i represent price/performance trade-off - citaj uporediti performanse i složenost ova dva pristupa).====

Koja su dva osnovna tipa multithreading-a?

-Block multithreading i Interleaved multithreading

Kako proceduralni konflikti uticu na SS procesor?

-Can not execute instructions after a branch in parallel with instructions before a branch

-Also, if instruction length is not fixed, instructions have to be decoded to find out how many fetches are needed

-This prevents simultaneous fetches

Razlike izmedju fine grain i coarse grain multithreadinga?

-Fine-grain multithreading—such as in a barrel processor -- issues instructions for different threads after every cycle, while coarse-grain multithreading only switches to issue instructions from another thread when the current executing thread causes some long latency events

Koji tip hazarda najviše usporava superskalarni procesor. Dati dijagram i objasniti.

-Valjda proceduralni hazardi

Pod kojim uslovima se kod In order issue, out of order completion, može zaustaviti pustanje instrukcija na izvršavanje?

--1) kada bi izdavanje posmatrane instrukcije dovelo do resursnog konflikta;

--2) kada upravo izdata instrukcija zavisi od rezultata instrukcija koje se još nisu kompletirale;

--3) kada rezultat posmatrane instrukcije može biti prebrisan rezultatom ranije izdate instrukcije kojoj treba više vremena da se izvrši.

Komponente za podršku multithreadinga u procesoru.

--1) višestruki specifikatori aktivnosti (npr. procesori sa više od jednog programskog brojača i više od jednog pokazivaca na stek);

--2) registarska podrška za višestruke kontekste;

--3) sinhronizacioni mehanizmi;

--4) mehanizmi za brzo smenjivanje niti.

Da li ima smisla za različite funkcije koristiti istu registarsku banku?

-Pa evo na ovo mislim da je odgovor da ima smisla, ako unutar jedne funkcije pozivas drugu preko zajednicke reg. banke mozes da prenošis argumente i rezultat funkcije.-

Kako se sprečava prepunjavanje bafera na prijemnoj strani kod PCI?

==Flow control protokol je deo transakcionog sloja. Flow control je point-to-point, a ne end-to-end protokol. Cilj ovog protokola je da spreči overflow u baferima koji implementiraju virtuelne kanale na prijemnoj strani nekog linka. Primalac šalje FC DLLP-ove (Flow Control Data Link Layer Packets) pošiljaocu kako bi ga obavestio o slobodnom prostoru svojim prijemnim baferima. Zagušenost jednog virtuelnog kanala ne sprečava pošiljaoca da prosleđuje pakete po drugim kanalima istog linka (jer su prijemni baferi kojima se implementiraju virtuelni kanali međusobno nezavisni).====

Sta je osnovna pretpostavka svih softverskih BPS?

--Da se ishod skoka može zaključiti iz konteksta programskog jezika visokog nivoa. Npr. za petlje sa proverom uslova na dnu veka je verovatnoca da ce do skoka doći jer su (dinamicke) statističke analize pokazale da se telo petlje obično izvrši više od jednog puta.