

Konputagailuen Arkitektura I

Autoebaluazio-Ariketak 3

1.- [1 puntu] Erantzun hurrengo galderei ikasgaien landutako BIRD oinarriko prozesadorea kontuan hartuz.

- Zergatik izaten dira aginduen exekuzioak azkarragoak kableatutako kontrol unitateetan, kontrol unitate mikroprogramatuetan baino? Horri aurre egiteko, zein da mikroprogramatutako kontrol unitateen abantaila nagusia, kableatuekin alderatuz?
- Agindu-multzoan $NEG R_i$ agindua gehitu nahi da. Agindu horrek eragigai bezala azaltzen den erregistroan gordetzen diren bitak banan-banan ezeztatzen ditu eta emaitza R_i erregistroan bertan uzten du (horretarako suposatuz UALan badagoela eragiketa bat eragigaiaren bitak ezeztatzen dituena):

$$R_i := \overline{R_i} \text{ \{bitak banaka ezeztatu\}}$$

Bere formatua honako hau da:



Idatz ezazu agindu hori exekutatze behar den mikroprograma zatia.

Prozesu Unitatean aldaketarik egin behar izanez gero, zehaztu ezazu argi eta garbi erantzun-orrietan emandako eskeman. Beste aginduen kontrolari dagokionez, ba al dago aldaketarik?

2.- [2 puntu] Konputagailu baten memoria sistemak ondorengo ezaugarriak ditu:

- Hitzak 8 bytekoak dira eta helbideratze unitatea bytea da.
- Alegiazko memoria:** 1 MB-ko memoria orriztatua 512 byteko orriekin. Helbideen itzulpena egiteko TLBa erabiltzen da. Honen atzipen-denbora 30 ziklokoa da hutsegitean eta 1 ziklokoa asmatzean. Hasieran, TLBa hutsik dago.
- Memoria nagusia:** 256 kB-koa da eta 4 modulu tartekatuz osatuta dago. Atzipen denbora 12 ziklokoa da (ziklo 1 tartekatze-bufferretik).

Ondorengo eskatzen da:

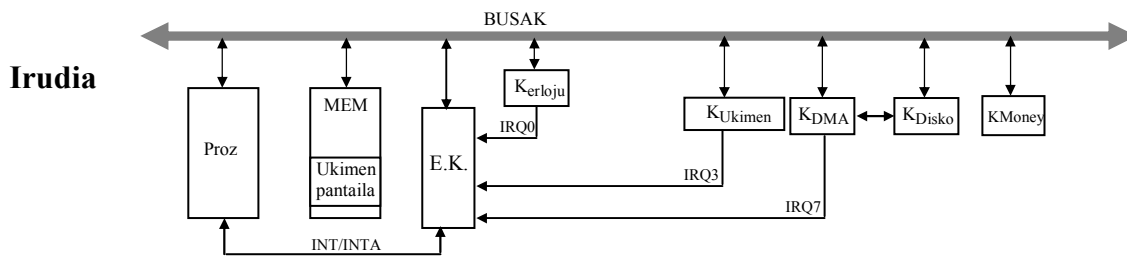
- Helbideen itzulpen eskema. Zenbat orri izan ditzake gehienez programa batek? Zenbat orri ditu memoria nagusiak? Zenbat posizio ditu orri taulak? Zein tamainakoa da posizio bakoitza? Zein da TLBko posizio bakoitzaren tamaina?
- Memoria nagusiaren helbidearen eskema eta memoria nagusiaren egituraren eskema. Zein da modulu baten tamaina bytetan?
- Konputagailu honetan ondorengo programa exekutatzen da:

```
for (i=2;i<64;i++)  
    x=C[4]+(A[i]*A[i+2]);
```

```
movi r1, #1296  
movi r3, #2  
load r2, [r1+680]  
beg: load r5, [r1]  
      load r6, [r1+16]  
      mul r5, r5, r6  
      add r10, r2, r5  
      add r1, r1, #8  
      addi r3, r3, #1  
      subi r4, r3, #64  
      bnz r4, beg
```

Programa 480 helbide logikotik aurrera metatua dago. A bektorea 1280 helbide logikoan hasten da eta C bektorea 1944 helbide logikoan. Bai aginduak eta bai bektoreen osagaiak hitz batetakoak dira. x eta i (begiztako indizea) aldagaiak makinako erregistroetan gordeta daude (ikusi mihiztadura lengoaiako kodea).

4.- [2,5 puntu] Fakultateko lehen solairuan **jolasteko makina** bat kokatu nahi da. Makina horretan ukimen pantaila bat erabiltzen da jolasari aginduak emateko. Guzti hori kontrolatuko duen sistema diseinatu nahi dugu. Horretarako, sistemak irudian azaltzen diren gailuak ditu.



Periferikoen kontroladoreen ezaugarriak honako hauek dira:

KMoney: Diruarentzako zirrikituaren kontroladorea da. Badauka egoera-erregistro bat (REGO_KMONEY) 1 balioa hartzen duena 50 zentimoko txanpon bat zirrikitutik sartzen dela detektatzen duenean. Erregistro horrek automatikoki 0 balioa hartzen du txanpona irensten ala kanporatzen duenean sistemak. Suposa ezazu bestelako txanponak ez dituela onartzen (edo ez dira zirrikitutik sartzen edo itzultzen dira). Kontroladore hau **inkestaz** sinkronizatzen da.

KUkimen: Ukimen pantailaren kontroladorea da. Erabiltzaileak pantailako punturen bat ukitzen duen bakoitzean etena eskatzen du. 3 byteko datu erregistro bat du (RDAT_KUkimen), pantailaren bereizmena 800*600 delako. Datu erregistro horretan erabiltzaileak sakatu duen tokiko erdiko pixela adierazten da. Kontrol erregistro bat ere badauka (RKON_KUkimen) eta bertan STROBE bat egin behar da kontroladore honek eteten duen bakoitzean.

K_DMA: DMA kontroladorea da. Memoriaren eta disko-unitate baten artean transferentziak egiteko erabiltzen da. Honako erregistro hauek ditu:

- **Kontrol-erregistroa** (RKON_KDMA): erregistro honetan lekua idaztean transferentzia hasiko da. (Automatikoki jartzen da 0an.).
- **Helbide-erregistroa** (RHEL_KDMA): transferituko den blokearen hasierako helbidea.
- **Luzera-erregistroa** (RLUZ_KDMA): transferituko den blokearen luzera bytetan.
- **Egoera-erregistroa** (REGO_KDMA): transferentzia amaitzean, erregistro honek adierazten du transferentzia ondo joan den (1) edo erroreren bat gertatu den (0).

K_DISKO: Diskoaren kontroladorea da. Transferentzia bete ahal izateko kontroladore hau hasieratu behar da. Suposatuko dugu *KdiskoProgramatu()* izeneko errutina dugula hasieratze hori egiteko.

Beste kontroladore guztiak (erlojuarena, teklatuarena eta etenena) ikasgai landutakoak dira. Ukimen pantaila memorian mapeatuta dago. Diruaren zirrikituaren kontroladorea izan ezik, inkestaz kontrolatzen dena, beste kontroladore guztiak etenen bidez kudeatzen dira.

Sistemaren **funzionamendua** honako hau izan beharko da.

Sistema martxan jartzean eta partidak jolasten inor ez dagoen bitartean, sistema argazki sorta bateko argazkietako bat ukimen pantailan erakusten egongo da. Argazki desberdin bat erakusten da 20 segundoero. Guztira KOP argazki desberdin daude eta horietako bat erakusteko **argazki_erakutsi(zenbaki)** errutina dago. Errutina horri parametro bezala erakutsi nahi den argazkiaren zenbakia pasatzen zaio. Segida logikoa hau da: 0 argazkitik hasi azkeneraino, 0 argazkira itzuli segidari berriz hasiera emateko eta horrela behin eta berriz.

Baina diru-zirrikituaren kontroladoreak 50 zentimoko txanpon bat sartu dela sumatzen duenean, 10 minutuko iraupena duen partidari hasiera ematen zaio. Horretarako, **partida_hasi()** errutina dago eta errutina horrek txanpona irenstea ere badakar. 10 minutu horietan zehar, erabiltzailea pantailako puntu desberdinetan ukitzen arituko da, azalduko diren iratxo batzuei tiro egin nahian. Tarte horretan, pantaila ukitzen den bakoitzean deitu behar zaio inplementatuta dagoen **posizioa_bidali(lerro, zutabe, &puntuazio)** errutinari. Errutina horri pasatu behar zaizkio ukitutako erdiko pixelari dagozkion lerroa (0 eta 599 arteko balio bat) eta zutabea (0 eta 799 arteko balio bat) eta eguneratuta itzultzen du ordura arte jokalaria lortutako puntuazioa, tiro egiterakoan lortutako asmatze edo hutsegitean araberak. Partiden hasieran puntuazioa 0 da.

Hamar minutuak pasatutakoan, partida amaitzen da eta memorian gordetzen da bertan lortu den puntuazioa **puntuazioa_gorde(partida_zkia, puntuazio)** errutinaren bidez. Errutina horri pasatzen zaizkio zenbatgarren partida izan den eta bertan lortutako puntuazioa. Puntuazio horiek memorian gordetzen dira PUNTHEL helbidetik aurrera eta bakoitzak 12 byte okupatzen ditu. Horren ondoren, sistema minutu batez zain geratzen da ea 50 zentimoko beste txanpon bat sartzen ote den. Hala gertatu ezean, berriro argazki segidako argazki desberdin bat erakusten jarriko da 20 segundoero.

Sisteman ehungarren partida bukatzen denean, lortutako puntuazioei buruzko informazio guztia memoriatik disko gogorrera kopiatu behar da DMA erabiliz. Transferentzia hori programatzen denean, pantailara ohar bat bidaltzen da *kopia_oharra_bidali()* errutinaren bidez. DMA transferentzia amaitzean, erroren bat gertatu bada, sistema behin eta berriz saiatuko da transferentzia egiten, inolako errorerik gertatzen ez den arte (sinplifikatzeko, suposatuko dugu transferentzia zuzen bukatuko dela argazkiak erakutsi gabe dagoen minutuan zehar). Dena den, 50 zentimoko txanpon bat sartzen bada DMA egiten ari den bitartean, ez zaio kasurik egingo eta txanpon hori itzuliko da *dirua_itzuli()* errutinaren bidez. DMA transferentzia zuzen bukatzen bada, sistemak kopia egiten bukatu duela adierazten du *kopia_amaituta()* errutinaren bidez eta lortzen diren puntuazio berriak PUTNHEL helbidetik aurrera gordeko dira ordutik aurrera. DMAren sinkronizazioa etenen bidez egin behar da.

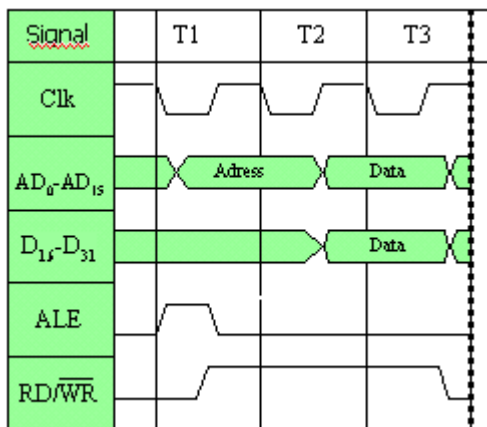
Hurrengoa eskatzen da: Idatzi lengoaia algoritmikoan beharrezkoak iruditzen zaizkizun zerbitzu-errutina guztiak eta programa nagusia. Komentatu ariketaren ebazpenerako egiten duzun edozein suposaketa. Kontuan hartuko da sistemaren portaera automata batez adieraztea.

5.- [0,7 puntu] Erantzun ondorengo galderak:

- a) Zergatik ez dira komunikatzen zuzenean periferikoak CPUarekin? Zer erabiltzen dute horretarako?
- b) Zer egiten dute `Eoi()` eta `Iret()` funtzioek? Zein da bietatik erlojuaren zerbitzu errutinan erabiltzen ez dena? Zergatik?
- c) Egin ezazu PCko eten bektorearen irudi bat. Zertarako erabiltzen da eten bektorea? Zein da eten bektorean teklatuaren zerbitzu errutinari dagokion posizioaren helbidea?

6.- [1 puntu] Irudian klasean aztertutako protokolo bat erabiliz burututako irakurketa baten kronograma ageri da. Erantzun ondorengo galderak:

- (a) Ze protokolo mota da eta zeintzuk dira bere ezaugarri garrantzitsuenak? Zeintzuk dira bere abantaila eta desabantaila nabarmenenak ikasitako beste protokolo batzuekin alderatuz?
- (b) Erloju maiztasuna 500 MHz bada, zein da lortzen den banda zabalera?
- (c) Zenbat denbora behar da 20 GB transferitzeko?



7.- [0,5 puntu] 1 GHz-eko erloju maiztasuna erabiltzen duen konputagailu bateko prozesadoreak nozitzen duen gainkarga kalkulatu nahi dugu sarrera/irteerako eragiketa bat egiten duenean etenen bidez disko gogor batekin. Etenen bidezko transferentzia bakoitza burutzeko, etenaren detekzioa barne, 1000 erloju-ziklo behar dira. Disko gogorak datuak blokeka transferitzen ditu, bloke bakoitza 16 hitzekoa da (hitz bakoitza 64 bitekoa izanik) eta bere funtzionamendu abiadura 32 MB/s-koa. Diskoak modu jarraituan egiten du lan eta bloke bat prest duen bakoitzean eteten du prozesadorea.

8.- [1,5 puntu] Erantzun aritmetikoei buruzko hurrengo galderak.

- a) Adieraz ezazu pausoz pauso nola lortzen den $A = 11$, $B = -7$ zenbakien arteko biderketaren emaitza, “batu edo jauzi” algoritmoaren bidez eta biderkagaiak 2 oinarriko Booth algoritmoaz birkodetarik. Eragigaiak 2 oinarrian eta oinarriko osagarrian adierazita daude, ahal den bit-kopururik txikienarekin.
- b) Egin ezazu $C = 23$ eta $D = 3$ zenbakien arteko zatiketa (C/D) berriztapenarekin. Eragigaiak 2 oinarrian adierazita daude, ahal den bit-kopururik txikienarekin. Zein da ateratzen den hondarra?
- c) Irudika ezazu 12 biteko *Carry-Select* batugailu baten eskema, kontuan izanik 3, 4 eta 5 biteko blokeak erabili behar dituzu ondoz ondoren. Adieraz ezazu batugailu horretan nola egiten den honako zenbaki hauen batuketa: $Z1 = EA3_H$, $Z2 = 15D_H$. Zenbatekoa da atzerapena? Eta kasurik txarrean? Justifika itzazu zure erantzunak.