



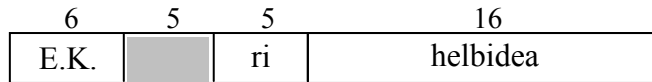
Konputagailuen Arkitektura I

Autoebaluazio-Ariketak 4

1.- [1 puntu] Erantzun hurrengo galderari ikasgaiaren landutako BIRD oinarritzko prozesadorea kontuan hartuz.

- a) Marraz ezazu kontrol-unitate mikroprogramatuaren μR erregistroa eta adieraz ezazu bere eremu bakoitzaren funtzionalitatea.
- b) Prozesadore horren agindu multzoa zabaldu nahi dugu agindu berri bat sartzeko: “`addp ri, ALDAGAIA`”. Agindu horrek `pc` erregistroan kargatu behar du `ri` erregistroaren edukiaren eta aginduan adierazitako aldagaiari dagokion memoria-posizioaren edukiaren arteko batura; hau da:
`pc := ri + MEM[helbidea]`

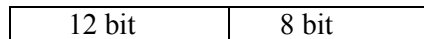
Bere formatua honako hau da:



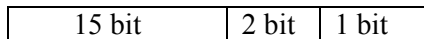
Idatz ezazu agindu horren exekuzio faseari dagokion mikroprograma zatia. Prozesu Unitatean aldaketarik egin behar izanez gero, adieraz ezazu eskematikoki.

2.- [2 puntu] Konputagailu baten memoria atzitzeko helbideek honako egitura hau dute:

- **Alegiazko memoria** orritzatua, helbide logikoak 20 bitekoak izanik:



- **Memoria nagusi** tartekatua, helbide fisikoak 18 bitekoak izanik:



Memoria hierarkia atzitzeko, honako denbora hauek behar dira:

TLB: hutsegitean 20 ziklo, asmatzean 1 ziklo

MN: 11 ziklo (1 ziklo tartekatze-bufferretik)

Konputagailu honetan ondorengo programa exekutatzen da:

for (i=0; i<1000; i++)

`B[i]=A[i]*A[i+2];`

begizta:

```

movi r1, #0
movi r2, #999
load r3, A[r1]
load r4, A[r1+4]
mul r3, r3, r4
store r3, B[r1]
add r1, r1, #2
subi r2, r2, #1
bge r2, begizta
    
```

Programa 0 helbide logikotik aurrera metatua dago. A bektorea 1024 helbide logikoan hasten da eta B bektorea 3040 helbide logikoan. Bai aginduak eta bai bektoreen osagaiak hitz batetakoak dira. Orri-taulako informazioa ondorengoa da:

Orri logikoa:	0	4	5	7	11
Orri fisikoa:	1	3	6	0	4

Erantzun ondorengo galderak:

- a) Helbideen itzulpen eskema. Zein da programa batek izan dezakeen tamaina maximoa? Zenbat orri logiko izan ditzake gehienez programa batek? Eta zein da memoria nagusiko orri fisiko kopurua?
- b) Itzuli programak begiztaren lehen iterazioan sortzen dituen memoriako helbide guztiak eta kalkulatu helbide bakoitzerako memoria sistema atzitzeko behar den denbora (memoria atzitzeko denbora eta helbideak itzultzeko denbora). Erabili ezazu taula bat egiten dituzun pausu guztiak bertan garbi azalduz.
- c) Kalkula ezazu programa osoko helbideak itzultzeko behar den denbora. Zertarako erabiltzen da TLBa itzulpen prozesuan?

3.- [0,5 puntu] Egin PCko eten kontroladorearen irudi bat. Zeintzuk dira bere erregistroak? Zertarako erabiltzen da erregistro bakoitza? Azaldu argi eta garbi zeintzuk diren PC batean eten baten tratamendurako jarraitzen diren pausoak. Zertarako erabiltzen dira INT eta INTA seinaleak?

4.- [1 puntu] Gure PCari sarrerakoa eta irteerakoa izan daitekeen gailu simple bat konektatu nahi diogu. Gailu hori inkestaz tratatuko da. Makina lengoaia mailan ikus daitezkeen 4 erregistro ditu, memorian mapeatu gabekoak. Erregistro horiek egoera erregistroa (283H portua), indize erregistroa (280H portua), datu erregistroa (281H portua) eta kontrol erregistroa (282H portua) dira. Indize erregistroa egotearen arrazoia zera da: kontroladoreak 15 byteko memoria txiki bat du, pantailaren kurtsorearen kontroladorearen antzera. Byte bakoitza zenbaki batek identifikatzen du. Zehazki, 4 bytea konputagailuan datuak sartzeko erabiltzen da eta 6 byteak konputagailuko datuak ezagutzera emateko balio du (irteerakoa da).

Egoera erregistroak 1 balio du 4 bytean balio bat sartu bada eta 2 balio du kontroladorea prest baldin badago 6 bytean balioen bat ateratzeko. Datu erregistroa erabiltzen da kanpotik datozen datuak eskuratzeko edota kanpora bidali behar diren datuak ezagutzera emateko. Datu erregistroa atzitzen den bakoitzean kontrol erregistroko 1 bitan STROBE segida bat egin behar da.

C lengoian errutina hauek inplementatzea eskatzen da:

- a) Kanpotik datorren balio bat jasotzeko inkesta egiten duen errutina; balio hori DATUIN aldagai orokorrean uzten du, ***void Inkesta_Irakurketa ()***.
- b) Behar den STROBE errutina, ***void Strobe ()***.

OHARRA: C kodea argitzeko, adieraz ezazu zer egiten duen idazten duzun kode-lerro bakoitzak. Ariketa hau egiteko, errutina hauek dituzu:

```
void Iret();                                void Eoi();  
unsigned char InPort (portua);  void OutPort (portua, balioa);  
unsigned char IrakurByteFis (unsigned short Seg, unsigned short Desp);  
void IdatzByteFis (unsigned short Seg, unsigned short Desp, unsigned char kar);  
unsigned short IrakurHitzFis (unsigned short Seg, unsigned short Desp);  
void IdatzHitzFis (unsigned short Seg, unsigned short Desp, unsigned short hitz);
```

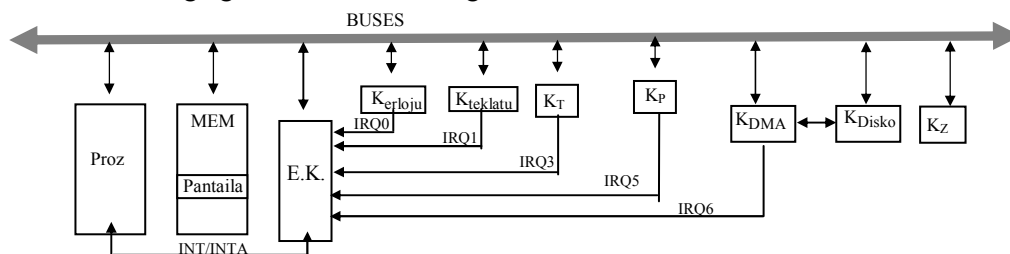
5.- [0,5 puntu] Konputagailu bateko prozesadoreak 2 GHz-eko erloju batekin egiten du lana eta prozesadoreak disko gogor batekin sarrera/irteerako eragiketa bat DMA bidez egiten duenean nozitzen duen gainkarga kalkulatu nahi dugu. Disko gogorrak datuak blokeka transferitzen ditu; bloke bakoitza 32 kB-koa da. Diskoaren funtzionamendu abiadura 32 MB/s da. DMA bidezko transferentzia bati hasiera emateko errutinak 4000 erloju-ziklo behar ditu, eta DMA kontroladorearen zerbitzu errutinak, berriz, 2000 ziklo. Zenbatekoa da sistemaren gainkarga?

6.- [1 puntu] Erantzun busei buruzko ondorengo galderak:

- a) Azaldu zeintzuk diren bus zatitu baten ezaugarriak: zenbat fase bereizten dira bus zatitu batean? Zertan datza bakoitza? Zein da fase bakoitzari hasiera ematen diona? Egin bus honen funtzionamendua azaltzen duen eskema bat.
- b) 2 MB/s banda zabalera duen bus sinkrono bat erabiliz 2 GB transferitu nahi ditugu. Zenbat denbora beharko du transferentzia honek? Gauza bera gertatuko litzateke busa erdisinkronoa ala asinkronoa balitz?

7.- [2,5 puntu] Bezeroentzat arretra zerbitzurako bulego batean kontrol sistema bat diseinatzea eskatu digute. Sistema honen helburua pertsonak bulegora iristen diren orden berdinean zerbitzatuak izatea da ilara bat erabiltzeko beharrik gabe. Ideia honekin, bulegoan leihatila bat eta txartel zenbakituak ematen dituen makina bat jarri dira.

Sistema honetan ondorengo gailuak aurki ditzakegu:



Erloju, teklatu eta etenen kontroladoreak ikasgaien ikusitakoak dira. Sistema honetako **teklatuaren** sinkronizazioa **etenen bidez** izango da. Pantaila memorian mapeatuta dago PANT_HELB helbidetik aurrera. Gainerako gailuen ezaugarriak ondorengoak dira:

K_Z: Txartel zenbakituak ematen dituen makinaren kontroladorea da. Bezeroak leihatilan zerbitzatuak izateko zenbaki bat esleitzea nahi duenean pultsadore hau sakatu beharko du eta makinak txartel bat emango dio hurrengo zenbakiarekin. Pultsadorea sakatzen denean kontroladorearen egoera-erregistroak (REGO_K_Z) bat balioa hartuko du. Makinak bi datu-erregistro dauzka eta erregistro hauetan adierazi behar da zein den eman behar duen hurrengo zenbakia (RDAT1_K_Z erregistroan) eta zein ordu den (RDAT2_K_Z erregistroan). Makinak txartela inprimatu dezan esleitutako zenbakiarekin eta orduarekin (RKON_K_Z) kontrol-erregistroan 1 balioa idatzi beharko da eta une horretan egoera-erregistroak berriro 0 balioa hartuko du. Gailu honen sinkronizazioa **inkesta** bidez izan beharko da.

K_P: Leihatilan dagoen pultsadore baten kontroladorea da. Leihatilaz arduratzen den langileak sakatzen duenean etengo du leihatila libre geratu dela adieraziz.

K_T: Txartel zenbakituak irakurtzen dituen irakurgailuaren kontroladorea da. Irakurgailuan txartel bat sartzean eteten du. Bi datu-erregistro ditu (RDAT1_K_T eta RDAT2_K_T) txarteletik irakurritako informazioa gordeko dutenak: txartelari esleitutako zenbakia eta txartela inprimatu deneko ordua. Gailu honen eten bat onartzen den bakoitzean STROBE sekuentzia bat egin behar da (RCON_K_T) kontrol-erregistroan.

K_{DMA}: DMA kontroladorea da eta memoriatik diskora transferentziak burutzeko erabiltzen da. Ondorengo erregistroak ditu::

- **Kontrol-erregistroa** (RKON_KDMA): 1 balioa idatzi behar da erregistro honetan transferentzia has dadin (0ra automatikoki jartzen da).
- **Helbide-erregistroa** (RHEL_KDMA): transferitu behar den blokearen hasierako helbidea.
- **Luzera-erregistroa** (RLUZ_KDMA): transferitu behar den blokearen luzera bytetan.
- **Egoera-erregistroa** (REGO_KDMA): 0 balioak azken eragiketa ez dela ongi bukatu adierazten du.

K_{DISCO}: Diskoaren kontroladorea da. Transferentzia bat egin nahi denean kontroladore hau ere programatu behar da, horretarako *KDiskoProgramatu()* errutina erabiltzen delarik.

Sistemaren **funtzionamendua** ondorengoa da:

Goizero, bulegoa ireki aurretik, langile batek sistema hasieratuko du txartel-zenbakiak 1 baliotik has daitezten ematen.

Bulegoa irekita dagoen denbora tartean, bezeroek leihatilan zerbitzatuak izan nahi badute bulegora sartzean txartel zenbakituak ematen dituen makinako pultsadorea sakatu beharko dute. Makinak ondoz ondoko zenbakiak emango ditu pultsadorea detektatzen duen heinean.

Leihatilako langileak libre dagoenean bertan duen pultsadorea sakatuko du. Une horretan leihatila gainean dagoen pantaila batean mezu bat aterako da zerbitzatu den hurrengo zenbakia zein den adieraziz, horretarako **idatz_pant(zenbakia)** funtzioa dagoelarik.

Leihatilari esleitu eta pantailan agertzen den zenbaki hori 15 segundotan gehienez izango da baliozkoa. Denbora tarte horretan, zenbaki hori duen txartelaren jabea leihatilara hurbildu eta langileari txartela eman beharko dio honek txartel irakurgailuan sar dezan. Baina 15 segundo horietan zehar ez bada txartelik sartzen irakurgailuan ez delako inor hurbildu, leihatilari hurrengo zenbakia esleituko zaio eta pantailan aterako da (zerbitzatu gabeko zenbakiak dauden bitartean). Ez bada zerbitzatu gabeko zenbakirik geratzen sistema zain geratuko da banatutako zenbaki berriak egon artean.

Zerbitzatzen den bezero berri bakoitzeko memorian ondorengo informazioa gordetzen da: esleitutako txanda zenbakia, zenbaki hori zein ordutan esleitu zaion eta bezeroa zein ordutan zerbitzatu den —ordua, ORDUA aldagai orokorrean dago—. Informazio honek guztiak 20 byte okupatzen ditu eta memorian idazteko *memidatzi (zenbakia, ORDU_{Aesleitu} ORDU_{Azerbitz})* funtzioa daukagu. Funtzioa arduratuko da zein memoriako helbidetan idatzi behar duen kontrolatzeaz: zerbitzatutako lehenengo bezeroari dagokion informazioa HELB_INF helbidean idatziko du eta ondorengo informazio guztia helbide horretatik aurrera, jarraian, metatuko da.

Orduaren tratamendurako ondorengo errutina hauek daukagu: *OrduaHasieratu()*, ORDUA aldagaia hasieratzeko sistemako ordua erabiliz eta *OrduaEguneratu()*, ORDUA aldagaia segundo batean eguneratzeko.

Estatistikak egiteko helburuarekin, iluntzean bulegoa ixten denean, langileak D tekla sakatuko du eta memorian egunean zehar gorde den informazio guztia diskora transferituko da DMA erabiliz. DMA transferentzia bukatzean erroreren bat gertatu dela detektatzen bada transferentzia behin eta berriz hasieratuko da, transferentzia ondo burutzen den arte. Suposatuko dugu transferentzia hurrengo goiza iritsi aurretik zuzen bukatuko dela. Transferentzia ongi bukatzean, programa bukatuko da eta sistema eragileari itzuliko zaio kontrola.

Ondorengoa eskatzen da: idatzi lengoia algoritmikoan beharrezkoak iruditzen zaizkizun zerbitzu errutina guztiak eta programa nagusia. Azaldu ariketa ebazterakoan egiten duzun edozein suposaketa.

8.- [1,5 puntu] Hau kalkulatu nahi dugu: $X=(A/B-C)*D$. Kalkuluak birako osagarrian egin nahi dira, behar diren gutxieneko bitekin, kasu zehatz honetarako: $A=27$, $B=6$, $C=-7$, $D=-12$.

- Zatiketa berriztapenik gabekoa da.
- $A/B-C$ kenketa bukaeran RC Arik gabeko CSA batek egiten du.
- Biderketa egiteko 2 oinarria duen Booth erabiltzen da.

Halaber, galdera hauei erantzun:

- a) *Overflow* gertatzen al da?
- b) Zenbat da zatiketaren hondarra?
- c) Kasu zehatz honetarako, biderketa modu eraginkorragoren batean egiteko bestelako modurik bururatzen al zaizu?
- d) Zenbat bitekoa izango da X?