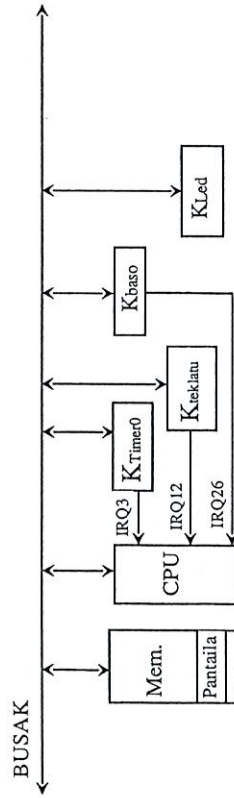


1. (8 puntos)

Nintendo Dsaren hardwarean oinarrituta, **sagardoa zerbitzatzeko sistema automatiko** bat garatu nahi dugu. Sistemaren hardware egitura honako da:



Gailuen kontrolagailuen ezaugarriak honakoak dira: los siguientes:

→ **K_baso**: sagardoa zerbitzatzeko den lekuan basoa jartzen edo kentzen den detektatzen duen gailuaren kontrolagailua da. Aipatutako lekuan baso bat jarritakoan zein kentzerakoan eten eskaera bat sortzen du dagokion IRQ lerroitik. Bere egoera-erregistroan (**REGO_Kbaso**) Okoa gordeko du basoa bere lekuan badago eta 1ekoa basorik ez badago. Eten bat onartzen den bakoitzean STROBE sekuentzia bat gauzatu behar da bere kontrol-erregistroan (**RKON_Kbaso**). Kontrol-erregistroak sagardoa zerbitzatzeko den bitartean basoa blokeatzeko ere balio du. Horretarako, 1ekoa jarri behar da 5 bitean eta desblokeatzeko 0koa.

→ **Kled**: sagardoa zerbitzatzeko dena prest dagoela adierazten duen led berde baten kontrolagailua da. Pizteko 1ekoa jarri behar da bere kontrol-erregistroan **RKON_KLed** (Okoa jarri itzaltzeko).

Bestelako kontrolagailuak, tenporizadorearenak, ukimen pantailarena eta teklatuarena, etenen kontrolagailuarenaz gain, klasean ikusitakoak dira. Sistema honetan, **teklatuaren** sinkronizazioa **etenen** bidez egiten da.

Sistemaren funtzionamendua honakoa da:

Sistemak ez du sagardorik zerbitzatzeko baso bat bere lekuan jartzen ez den bitartean. Sistema ez zailtzearren suposatuko dugu sagardo kupelak egun osoa irauteko adina sagardo baduela eta sagardotegia itxita dagoenean beteko dela.

Baso bat dagokion lekuan jartzerakoan sistemak leda piztuko du (**LedaPiztu()** errutinen bidez). Horrela sagardoa zerbitzatzeko prest dagoela adierazten da. Erabiltzaileak ukimen-pantaila ukitzerakoan sagardoa zerbitzatzeko hasiko da (**SagardoaZerbitzatu()** errutinen bidez). 5 segunduz zerbitzatu du, lehenago erabiltzaileak ez badu ukimen-pantaila ukitzen. Hala izanen, une horretan bertan utziko dio zerbitzatzeari (**SagardoaZerbitzatuUtz()** errutinen bidez).

Sagardoa zerbitzatzeko ari den bitartean basoa blokeatu egin behar da sagardoa gal ez dadin. Horretarako **BasoaBlokeatu()** errutina erabili behar da. Behin sagardoa zerbitzatzeko amaitu egin denean, sistemak leda itzali egingo du, **LedaItzali()** errutinaz eta basoa desblokeatu egingo du **BasoaDesblokeatu()** errutinaz.

Zerbitzatu ondoren behar beharrezkoa da basoa bere lekutik kentzea sagardo gehiago eskatu aurretik. Ez da sagardorik zerbitzatzeko sagardoa jaso eta sistematik erretiratu ez den baso batean.

Honakoa eskatzen da:

- (1,5 puntu) Sistemaren diseinua egin ezazu egoera-makina edo automata baten bidez.
- (1 puntu) Idatz ezazu beharrezkoa den kodea etenen kudeatzaileak jakin dezadan eten bakoitzarekin zein zerbitzu errutina exekutatu behar duen.
- (4 puntu) Idatz lengoaia algoritimikoan programa nagusia eta beharrezkoak diren zerbitzu-errutinak. Suposatu tenporizadorearen kontrolagailuak segunduan 60 aldiz eten egingo duela (ez kezkatu parametroekin, 60 jarri eta listo).
- (1,5 puntu) Idatz ezazu **void BasoaBlokeatu()** eta **void BasoaDesblokeatu()** errutinak inplementatzen dituen kodea idatz ezazu.

Notai: Gogoratu pantailaren koordenatuak irakurtzeko *touchPosition* motako *PANT_DAT* aldagaila erabil dezakezula *touchRead* errutinen bidez.

```
void touchRead(touchPosition *PANT_DAT);
```

2. (2 puntu)

Demagun badaukagula erloju sinkrono bat duen sistema bat, non erlojuak 400 Mhz-ekoa den. Sistema honetan memoriaren irakurketak 2 erloju ziklo behar ditu. 8 helbide lerro eta datu eta helbidearako multiplexatutako 16 lerro ditu. Helbideratzea byteka egiten da.

- (0,25 puntu) Kalkulatu irakurketara beharko duen denbora segundutan.
- (0,5 puntu) Kalkulatu busaren banda-zabalera.
- (0,5 puntu) Zein da sistemaren helbideratze espazioa?
- (0,25 puntu) Adieraz ezazu ze seinale azaltzen den bus sinkrono batean eta ez asinkronoan.
- (0,25 puntu) Ze seinale erabiltzen dira bus asinkrono baten sinkronizazioan?
- (0,25 puntu) Ze seinale erabiltzen da multiplexatutako lerroetatik bidaltzen den informazioa uneoro zein den adierazteko?

AZPISISTEMEN ARTEKO KONEXIOA: BUSAK

KE

Busak

1

Sarrera: definizioak

- **Bus-zikloa:** bi osagairen artean datu baten oinarritzko transferentzia egiteko behar den denbora
 - Busaren oinarritzko eragiketa
 - Urratsak: busaren eskaera, arbitratzea, helbideratzea, transferentzia, errore-detekzioa, onarpena
- **Busaren zabalera:** datu-lerroen kopurua busean
- **Busaren banda-zabalera:** parametro honek adierazten du zenbat byte bidai daitezkeen busetik denbora unitatean
 - Adibidea: 1 Mbyte/s \rightarrow 10⁶ byte/s
 - Transmisio-abiadura: luzeraren, kontrol-logikaren... arabera.
- **Busaren protokoloa:** busera konektatuta dauden gailuek bete behar duten arau-multzoa komunikazio zuzena lortzeko

KE

Busak

3

Sarrera

- **Helburua:** konputagailuaren osagaiak elkartzen dituzten bideak aztertzea
- **Busa:** CPU, memoria eta sarrera/irteerako gailuak lotzen dituen seinale-lerro multzo bat da
 - **Helbide-lerroak:** memoria edo S/Iko portuen helbidea
 - **Datu-lerroak:** idatzi edo irakurri behar diren datuak
 - **Kontrol-lerroak:**
 - + egin beharreko eragiketa: irakurketa edo idazketa, ...
 - + transferentziaren kontrola: protokoloa eta denborizazioa
 - + arbitratzea: osagai batek baino gehiagok batera eskatzen badu busen kontrola (CPU, DMA, S/Iko periferikoak), nori eta noiz eman busen kontrola
- **Funtzionamendua:** osagaiak busaren kontrola lortu behar du eta transferentzia bete protokoloa jarraituz
- **Eragiketak:** irakurk./idazk. (datua/blokea), *Read-Modify-Write...*

KE

Busak

2

Sarrera: definizioak

- **Jabea (master) eta morroia (slave):**
 - Jabea: busaren jabea da eta transferentzia bat hasiera dezake
 - Morroia: gailu pasiboa eskaeren zain
 - Adibideak:

Jabea	Morroia	Eragiketa
CPU	Memoria	Aginduen bilaketa
CPU	S/I-ko gailuak	Transferentzia hasieratu
DMA	Memoria	Datuen transferentzia
- Erlazio dinamikoa: A gailua jabea izan daiteke transferentzia batean, baina morroia beste batean. Adibidez, DMA kontroladorea

KE

Busak

4

Sarrera: busen hierarkia

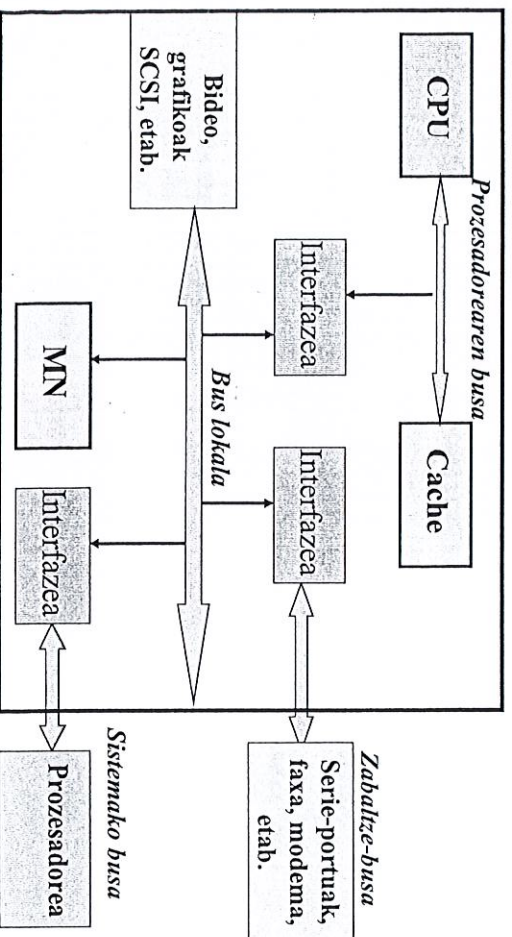
- **Arazoak** gailu asko konektatzen bada busera:
 - gailuen arteko seinaleen hedapenean atzerapen handiagoa
 - gailu bakoitzak ezaugarri desberdinak: funtzionamendu-abiadura, banda-zabaleraren behar desberdinak etab.
 - busaren ahalmena ase
 → bus bakarra: konputagailuaren ezaugarri ahulena
- **Irtenbidea:**
 - bus desberdinak erabiltzea hierarkikoki antolatuta, bakoitza bere zabalera eta abiadurarekin
 - ezaugarri berdinsuak dituzten osagaiak bus bera erabiltzen dute
 - CPU-ra hurbildu ezaugarri hobereak dituzten gailuak

KE

Busak

5

Sarrera: busen hierarkia



Prozesadore bakarreko sistema

KE

Busak

7

Sarrera: busen hierarkia

- **Barne-busa:** CPU barneko komunikazioa
- **Prozesadorearen busa:** CPU eta kanpoko cachearen arteko komunikazioa. Adibidez: P4ren busa 400 MHz-koa
 - Luzera txikia eta abiadura
 - Sistema bakoitzak bereak, espezifikoak (prozesadorearen leerroak)
- **Bus lokalak:** abiadura altuko busak azkarrek diren S/I-ko gailuentzat. Ezaugarri orokorreko busak izan daitezke, PCI adibidez, edo dedikatutak: IDE, SCSI, AGP, USB...
- **Zabaltze-busa :** abiadura txikiko S/I-ko gailuak konektatzeko (faxa, modema, serie-portua, ...). Adibidez: ISA, MCA
 - Osagai desberdin asko, banda-zabalerak desberdinak
- **Sistemako busa:** sistema bera osatzen duten prozesadore sistema desberdinak konektatzeko. Adibidez: VME

KE

Busak

6

Denborizazioa: transmisio-protokoloa

- Nola koordinatzen diren gertaerak busean transmisioa zuzena izan dadin
- Oinarritzko bi protokolo: **sinkronoa** eta **asinkronoa**
- **Protokolo sinkronoa:**
 - erraza gauzatzeko, baina malgutasun gutxikoa: gailuen abiadurak desberdinak dira eta geldoenaren abiadura hartu behar da guztientzat
- **Protokolo asinkronoa:**
 - transferentzia egokitu daiteke gailuaren abiaduraren arabera

KE

Busak

8

Bus sinkronoa

- Buseko erloju-seinale batek kontrolatutako transferentziak
- Transferentzia batek ziklo-kopuru osoa behar du
- Protokoloaren adibidea:
 - T_{AD} : erlojuaren lehen zikloko goranzko ertzetik helbidea egonkortzen den unera arte pasatzen den denbora maximoa
 - T_{DS} : irakurritako datuek hirugarren zikloko beharantzko ertza baino zenbat denbora lehenago egon behar dute busean, inork irakurri aurretik egonkorrak izan daitezten
 - T_M eta T_{RL} : denborek adierazten dute lehen zikloko beharantzko ertzaren ondoren zenbat denbora pasa daitekeen gehienera IO/M.L eta RD/WR.L seinaleak aktibatzen gabe.

KE

Busak

9

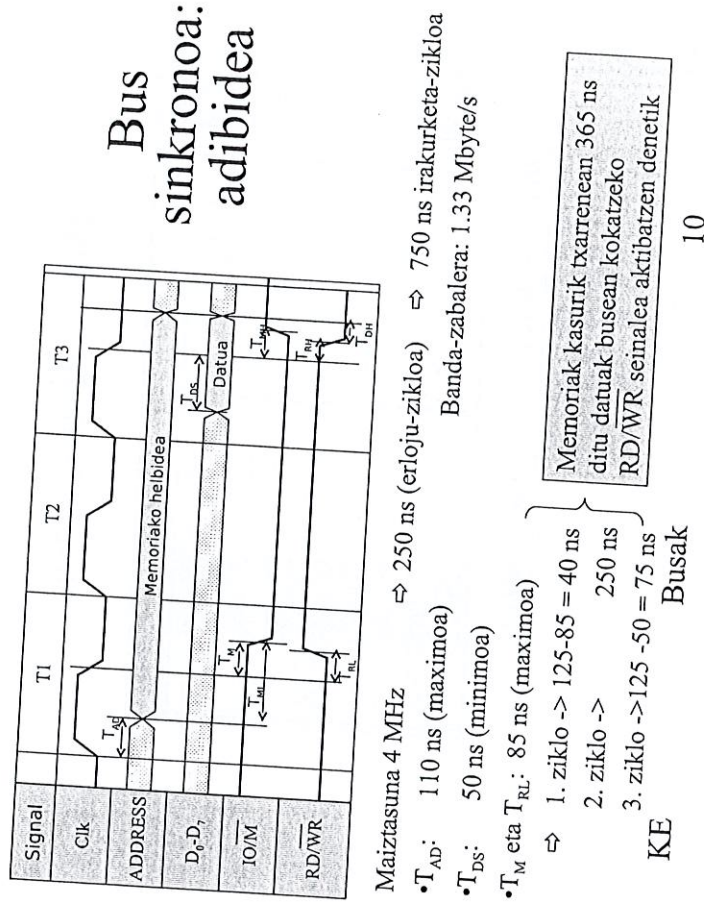
Bus sinkronoa

- **Busaren banda-zabalera handitzea:**
 - erloju-maiztasuna handitu
 - arazoak gailuekin → erantzun-denbora txikiagoa behar da
 - arazoa busean (*bus skew*): seinale guztiak ez dira busean agertzen une berean eta erlojuak denbora eman behar die seinale guztiei egonkorrak izateko
 - transferentziak blokeka egin
 - jabeak adierazten dio morroiari transferitu behar den byte-kopurua
 - morroiak, bus-zikloan byte bat itzuli beharrean, byte bana bidaltzen du erloju-ziklo bakoitzean adierazitako byte-kopurua bidali arte
 - lehen ikusitako adibidean, n byteko blokea irakurtzeko denbora $n+2$ izango litzateke $3n$ izan beharrean

KE

Busak

11



10

Bus erdisinkronoa

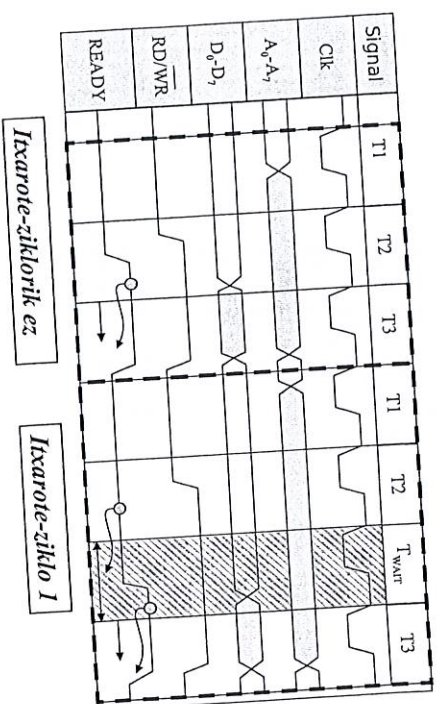
- Bus sinkronoaren ezaugarriak baina seinale bat gehiago: **READY** (baita **BUSY** edo **WAIT**)
- Gailu batek ezin duenean eragiketa bete aurreikusitako denboran, jabeari jakinaraziko dio **READY** seinalea ez aktibatuz → ziklo gehiago gehitzen dira: **ixarrote-zikloak** (*wait state*)
- Datuak prest daudenean, morroiak **READY** seinalea aktibatzen du
- Seinale honen bitartez protokoloa egokitu daiteke gailuaren abiadurara. Gehitutako ixarrote-zikloen kopurua beti osoa izango da
- Erloju-seinalea erabiltzen duten protokoloek ez dute lortzen emaitza onena: eragiketa batek 550ns behar baditu eta erlojuaren periodoa (zikloa) 250ns-koa bada, 3 erloju ziklo beharko dira eragiketa burutzeko (750ns). Eraginortasuna galtzen da.

KE

Busak

12

Bus erdisinkronoa

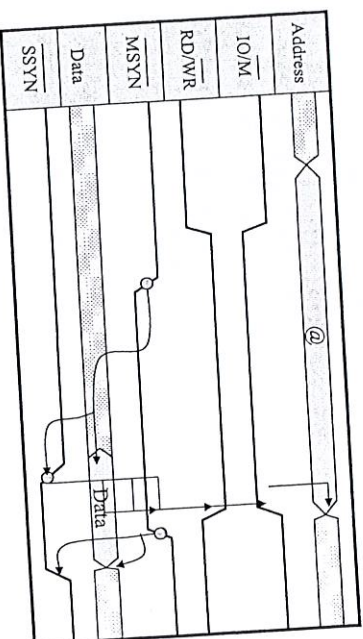


KE

Busak

13

Bus asinkronoa



KE

Busak

15

Bus asinkronoa

- Busak ez du erloju-seinalerik. Gailuen arteko komunikazioa elkaritzeketa-protokolo bat jarraituz egingo da (*handshake*). Horretarako bi seinale berriak izango dira: **MSYN** (*master synchronization*) eta **SSYN** (*slave synchronization*)
 - Jabeak helbide- eta kontrol-seinaleak aktibatzen ditu (adib. irakurtzeko)
 - Seinaleak egonkortzeko tarte bat utzi eta gero MSYN seinalea aktibatzen du, horrela adieraziz helbide- eta kontrol-lerroak egokiak direla
 - Denbora-tarte ezezagun bat eta gero, morroiak datuak ematen ditu eta SSYN seinalea aktibatzen da datuak prest daudela adierazteko
 - Jabeak datuak jaso eta MSYN desaktibatzen du
 - Morroiak SSYN desaktibatzen du

KE

Busak

14

Bus zatitua

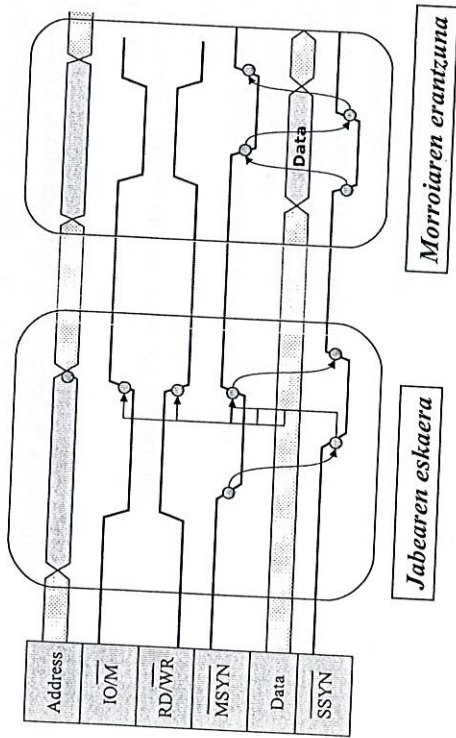
- Bi fase bereizten dira: jabearen eskaera eta morroiaren erantzuna
 - bi faseen arteko denbora beste transferentzia baterako erabiliko da
- Seinaleak MSYN eta SSYN (bus asinkronoa)
- Lehen fasea (adibidez. Memoriako irakurketa):
 - Jabeak helbidea bidaltzen du eta IO/M, RD/WR.L seinaleak aktibatzen ditu. Horrekin batera bere identifikadorea bidali behar du
 - Seinaleak egonkortzeko tarte eta gero, MSYN aktibatzen du
 - Morroiak SSYN aktibatzen du eta ondorioz jabeak seinaleak desaktibatu eta deskonektatu egiten da. Azkenik, morroiak SSYN desaktibatzen du
- Bigarren fasea: lehen morroia zena orain jabea da
 - Morroiak datuak prest dituenean, transferentzia hasieratzen du busean: datuak eta jabearen identifikadorea jarri eta SSYN seinalea aktibatzen du
 - Jabeak datuak jaso eta MSYN aktibatzen du
 - Morroiak SSYN desaktibatzen du, eta ondorioz, jabeak MSYN desaktibatzen du

KE

Busak

16

Bus zatitua

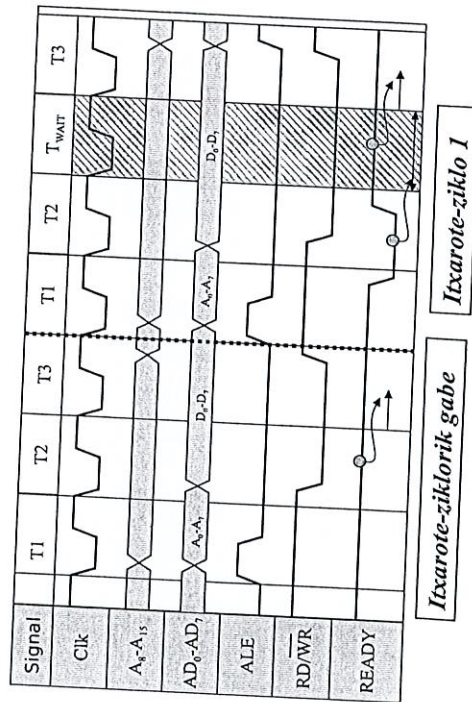


KE

Busak

17

Busen multiplexazioa



Bus erdisinkrono multiplexatuaren adibidea

KE

Busak

19

Busen multiplexazioa

- Lerro berak erabiltzea seinale-mota desberdinak bidaltzeko, adibidez, helbideak eta datuak
- busaren lerroak denboran multiplexatzen dira
- Seinale berria: **ALE** (*Address Latch Enable*), lerro multiplexatuak zertarako erabiltzen ari diren uneoro jakiteko
- Datu edo helbide-lerroen multiplexazioaren bitartez:
 - (a) helbideratze-ahalmen bera helbide-lerro gutxiagorekin, datu-lerroak multiplexatzen badira
 - (b) banda-zabalera handiagoa helbide-lerroak multiplexatzen badira
- Hardwarea sinpleagoa da, baina protokoloa motelagoa (ezin da informazio guztia paraleloan bidali)

KE

Busak

18

Busaren arbitratze-lana

- Bus batean osagai-jabe bat baino gehiago baldin badago, nork erabil dezake busa aldi berean behar badute? Nola kudeatzen dira lehentasunak busa eskuratzeko?
- Protokolo guztiak oinarizko 3 seinale erabiltzen dituzte:
 - Busaren eskaera** (*Bus Request*): busa eskuratu nahi duen gailuak aktibatzen du
 - Eskaeraren onarpenera** (*Bus Grant*): busaren arbitroak bidaltzen duen seinalea gailuari adierazteko busaren jabea dela
 - Busa okupatuta** (*Busy*): gailuak busa lortzen duenean seinale hau aktibatzen du busaren kontrola eskuratuz

KE

Busak

20

Bibliografia

- **Liburuak**
 - Arquitectura del PC (volumen II), M. Ujaldón
 - Organización y Arquitectura de Computadores, W. Stallings
 - Introducción a la Informática, A. Prieto et al.
- **Aldizkariak**
 - Byte, PC Actual, PC World, PC Plus, PC Magazine
- **Web orriak**
 - <http://www.techfest.com/hardware/bus/isa.htm>
 - <http://www.pcisig.com>
 - <http://www.agpforum.org>
 - <http://www.1394ta.org> (Fire Wire busa)
 - <http://www.usb.org>
 - <http://www.t10.org> (SCSI busa)
 - <http://www.t13.org> (IDE busa)
 - <http://www.scsi.org> (SCSI eta Serial Attached SCSI busak)

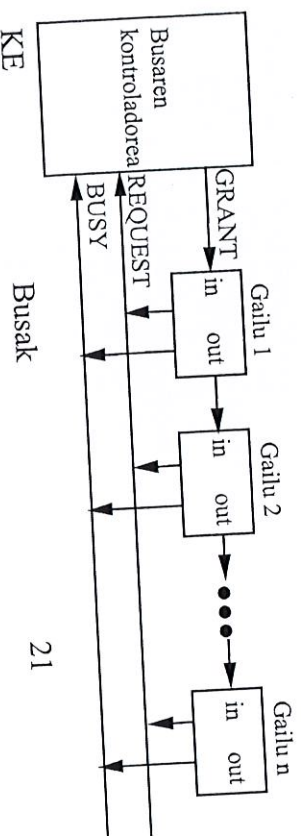
KE

Busak

22

Margarita-katea (Daisy-Chain)

- Gailuak *Bus Request* aktibatzen du → arbitroak *Bus Grant* aktibatzen du
- *Bus Grant* seinalea lehen gailura iristen da (*in*), eta honek:
 - ez badu eskaerarik egin, hurrengoari pasako dio → *out*
 - eskaera egin badu, katea moztu eta *Busy* aktibatuko du
- Transferentzia amaitzean *Busy* desaktibatuko du eta busa libre utzi
- Gailuen lehenetasuna finkoa: arbitrotik gertuen dagoenak lehenetasun handiena



Busak

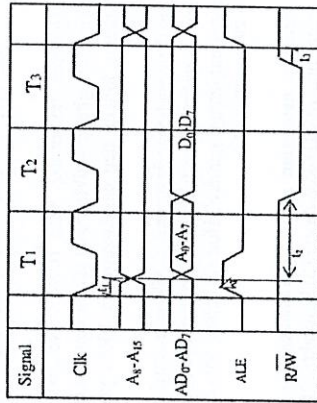
21

KE

1.- Ondoko diagraman sistema baten bus-zikloa ageri da memorian idazketa bat betetzeko. Diagrama horren arabera, adieraz ezazu:

- Eskematik ater daitezkeen komunikazio-protokoloaren ezaugarriak.
- Erlojuaren maiztasuna 2.5 MHz-ekoa bada, zein da memorian idazteko denbora maximoa R/W seinalea aktibatutik ondoren?
- Protokoloari gehituko zenizkiokkeen aldaketak memoria geldoango batekin lan egin ahal izateko.
- Memorian idazteko denbora maximoa protokoloa asinkronoa balitz.

t1 = 50 ns (maximoa)
t2 = 370 ns (maximoa)
t3 = 80 ns (minimoa)



2.- Osatu ezazu ondoko taula, X batez adieraziz seinale bakoitza zein bus-motetan azaltzen den.

Seinaleak/ Busak	SINKRONO MULTIPLIXATUA	ASINKRONO (bus zatitua)	SEMSINKRONO
ERLOJUA	X		X
WAIT			X
MSYN		X	
SSYN		X	
ALE	X		

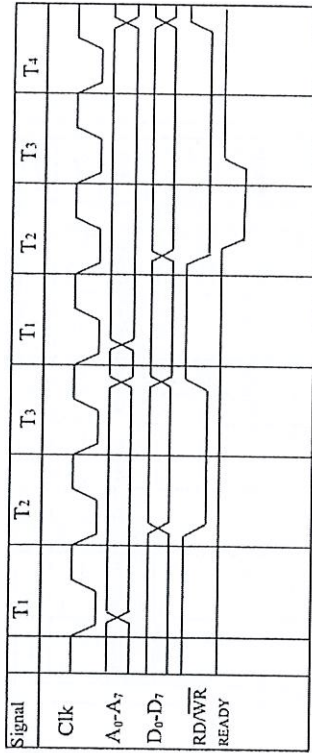
b) Azaldu modu laburrean (gehienez 3 lerroetan) zertarako balio duten ondoko seinaleak:

- WAIT (edo READY edo BUSY)
- ALE

3.- Bus sinkronoa eta 8 MHz-eko erlojua erabiltzen duen sistema batek 3 erloju-ziklo behar ditu edozein gailutatik irakurketa bat egiteko. Eraginikortasuna handitu asmoz, erlojua 9 MHz-eko beste batengatik aldatzen da. Baina honi esker busa erdisinkronoa bihurtzen da eta batez beste 3.5 ziklo (hau da, 3 ziklo gailu batzuentzat eta 4 ziklo motelagoak diren beste gailu batzuentzat) behar dira irakurketa bera egiteko.

- Hau ikusita, aipaturako aldaketa egiteak merezi du? Justifika ezazu.
- Edozein kasutan, modu laburrean aipa itzazu bus sinkrono eta erdisinkronoen arteko desberdintasunak.

4.- Honako kronogramak bi idazketa betetzeko sistema baten bi bus-ziklo azaltzen ditu.

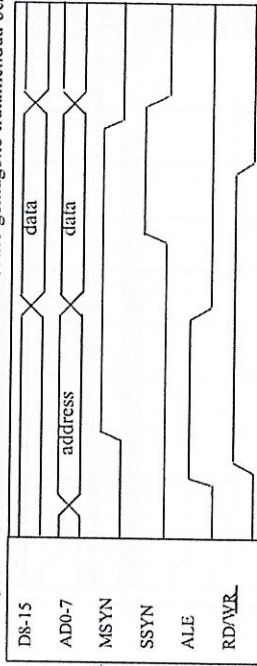


Ondokoa eskatzen da:

- Adierazi zein protokolo-motari dagokion.
- Erlojuaren maiztasuna 8MHz-koa bada, kalkulatu zenbat denbora behar den idazketa bat burutzeko kasurik onenean.
- Zein aldaketa egingo zenuke protokoloan 64 Kbyte helbideratzeko, ahalik eta lerro gutxienez erantsiz?

5.- Hurrengo irudiak klascan aurkeztutako bus-protokolo batcan irakurketa bat egiteko seinaleak eta urratsak adierazten ditu. Honakoa eskatzen da:

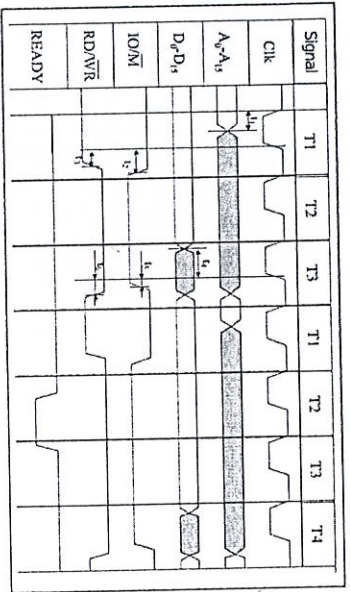
- Adierazi zein protokolo den eta zeintzuk diren bere ezaugarri aipagarrienak.
- Esan laburki zeintzuk diren bere abantaila eta desabantaila nagusienak.
- ALE seinalea kentzeak zer eragin izango luke protokoloaren eraginikortasuneari?
- Aipatu zer aldatuko zenuke eskaera bat baino gehiagoko tratamendua bermatzeko.



6.- 50 MHzko erloju-maiztasuna duen bus sinkrono batek 3 erloju-ziklo behar ditu 16 biteko irakurketa bat egiteko eta 4 erloju-ziklo idazketa bat egiteko.

- Zein da busaren banda-zabalera irakurketa? Eta idazketa?
- Zenbat denbora behar izango da 1 Mbyte irakurri nahi bada? Eta idatzi nahi bada informazio hori?

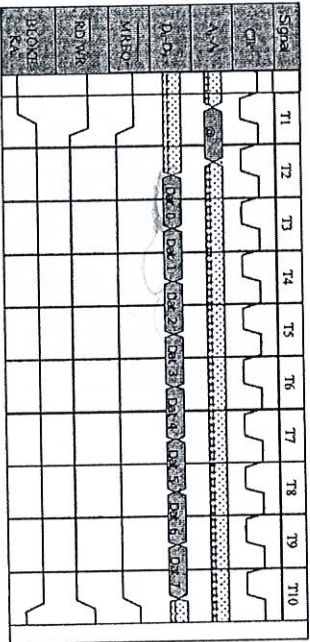
8.- Ondoko irudiak sistema batean bi irakurketa jarraian egitean gertatutakoa azaltzen du:



Handwritten notes: $10 \text{ MHz} \rightarrow 10 \cdot 10^6 \text{ Hz}$, $f = \frac{1}{T} \Rightarrow T = \frac{1}{f} = \frac{1}{10 \cdot 10^6} = 1 \cdot 10^{-7} = 100 \text{ ns}$

- Hau kontuan izanik, ondoko adierazi:
- Irudian oinarrituz, adierazi sistemaren komunikazio-protokoloaren ezaugarriak.
 - Zerituzak dira aieratutako datuek? Helbidetako sistema honetan?
 - Erlojuaren maiztasuna 30 MHzkoa bada, zein da busaren **banda-zabalera** kasurik onenean?
 - Nola handituko zenuke **banda-zabalera**, busaren ahalik eta aldaketak gutxien eginuz? Zein da banda-zabalera berrira?

8.- Sistema batek 10 MHzko erloju-maiztasuna duen bus sinkronoa du. Irudian ikus daitezkeen bezala, transferentziak blokeak egin daitezke eta modu honetan jarraian dauden 8 datu transferitzen dira.



- Sistemaren transferentzia gaitasuna handitu nahi dela eta, kostea kontuan izanik, datuen eta helbiden terroak multiplexatzea pentsatu da. Aldaketa honek transmisioa blokeak egiteko aukera galitza badaukar, honako eskatzen da:
- Lehen sistemaren banda-zabalera (transmisio normaletan eta blokekatuetan).
 - "Hobetuak" sistemaren banda-zabalera.
 - Blokekatu transferentziak kasuen %60an erabiltzen badira, aldatuak merkezi al du?

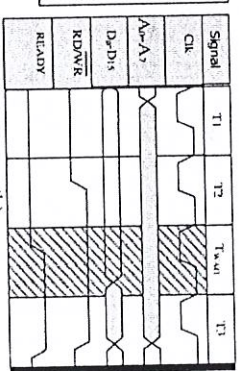
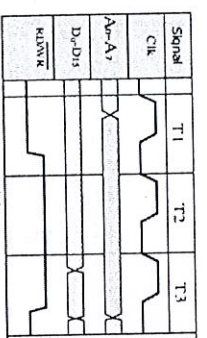
9.- Azal itazu gailu batek busa erabiltzeko jarraitu beharreko pausuetak daizy chain arbitrazzen erabiltzen denem. Suposa itazu 4 gailu eta beraien arteko gatazka gertatzen ez direla eta kalkulatu zenbat denbora behar duen arbitroak urrutien dagoen gailuak busa eskuratzeko, ondoko denbora kontuan hartuz:

- gailu guztiak 10 nseg behar dituzte REQ/UEST eta BUSY seinaleak aktibatzeko;
 - arbitroak 4 ns behar ditu GRANT seinalea aktibatzeko; gailuek ere 4 ns behar dituzte seinale bera aktibatzeko bus-estakera ez badute egin.
 - seinaleek buscan duen hedapen atzerapena 2ns-takoa da.
- Erlojuaren maiztasuna 500 MHzkoa bada, zenbat ziklo behar dira?

- 10.- Gure komputagailuan memoriako busa 100MHz-ko maiztasuna duen bus sinkrono bat da. Memoriako irakurketa bat egiteko 4 ziklo behar direla kontuan izanik, erantzun ondorengo galderak:
- Irakurketa lortzen den banda-zabalera 100Mbyte/s-koa bada, zein da datu-ousaren zabalera?
 - Zenbat denbora behar da busetik 5 Mbyte irakurti eta transferitzeko?
 - Sistema honetan 100ns-tako erantzun-denbora duen memoria bat sartzea nahi badugu, arazoren bat izango genuke? Horrela balitz, nola konponduko zenuke? Azaldu argi eta garbi zure erantzunak.

11.- (a) eta (b) irudietan memoriari irakurketa betetzeko bi bus ziklo desberdin ageri dira. (a) kasurako erloju-maiztasuna 16MHzkoa da eta (b) kasurako 32MHzkoa.

- Ze protokolo mota da irudi bakoitzeko? Azaldu protokolo hauen arteko desberdintasunak.
- Zein da banda-zabalera (a) kasurako?
- 250ns-tako erantzun-denbora duen memoria bat erabiltzea nahiko bagenu gure sistemari, zein protokolo aukeratuko zenuke?
- Memoriaren erantzun-denbora 50ns-takoa balitz, ze protokolo litzateke eraginkorragoa 16 biteko datuak bidaltzeko? Azaldu zure erantzunak.



Handwritten notes: $10 \text{ MHz} \rightarrow 10 \cdot 10^6 \text{ Hz}$, $f = \frac{1}{T} \Rightarrow T = \frac{1}{f} = \frac{1}{10 \cdot 10^6} = 1 \cdot 10^{-7} = 100 \text{ ns}$