

6. gaia:

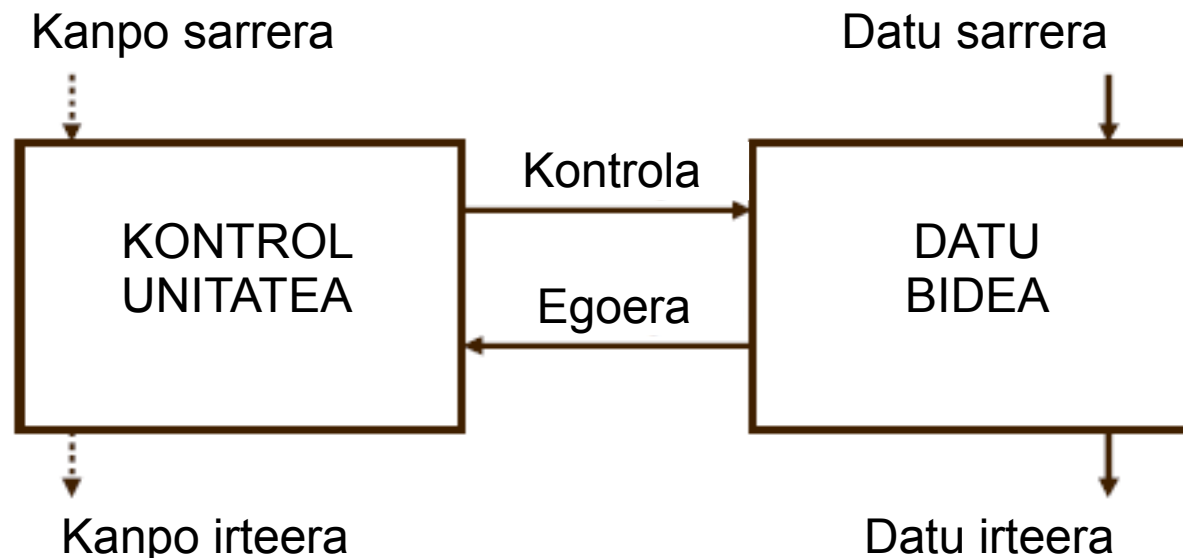
Sistema digitalen diseinu  
metodologiaren hastapenak

# Sistema digital konplexuen diseinua

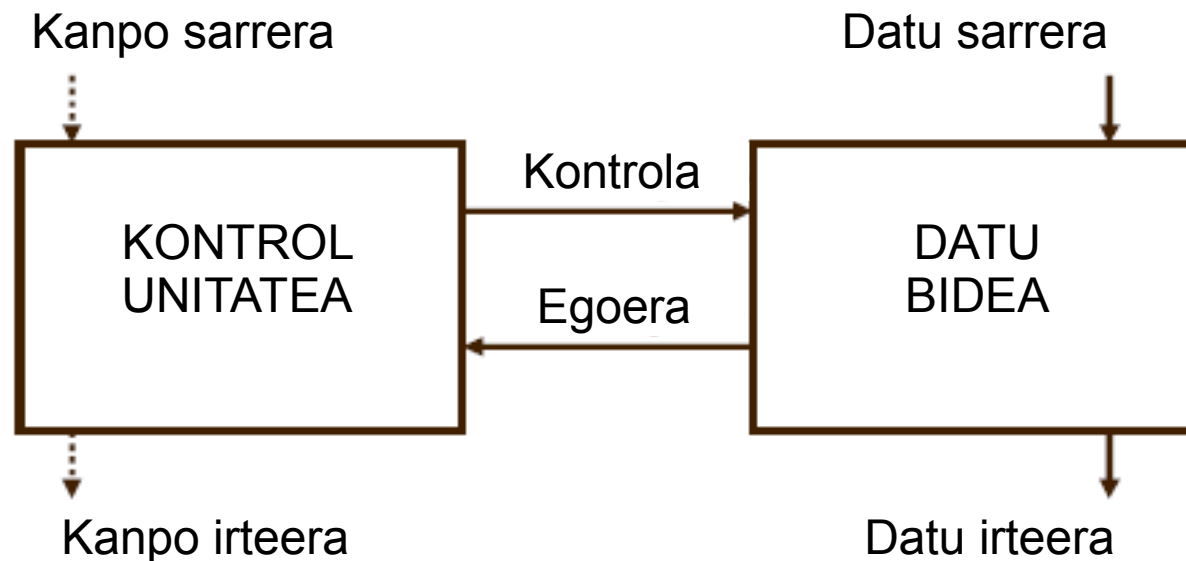
- Sistema digital konplexuak diseinatzekoan, bi ataletan banandu ditzakegu:
  - Datu bidea
  - Kontrol unitatea
- Datu bidea → Eragiketa aritmetiko, logiko eta desplazamendukoak egiten ditu
- Bere elementuak erregistroak, UAL (unitate aritmetico-logikoa), multiplexadoreak, konparadoreak, eta abar dira

# Sistema digital konplexuen diseinua

- Kontrol unitatea → Datu bide eragiketaren gaikuntza eta sekuentziazioa egiten du
- Datu bide elementuen gaikuntzarako kontrol seinaleak eragiten ditu
- Eragiketak noiz egiten dira erabakitzen du



# Sistema digital konplexuen diseinua



- Kontrol seinaleak jasotzen ditu datu bidek kontrol unitatetik, eta kontrol unitateari eragiketei buruzko informazioa (egoera) bidaltzen dizkio
- Kontrol sarrerak jasotzen ditu kontrol unitateak erabiltzailetik, eta eragiketen emaitzari buruzko informazioa bidaltzen dio

# Sistema digital konplexuen diseinua

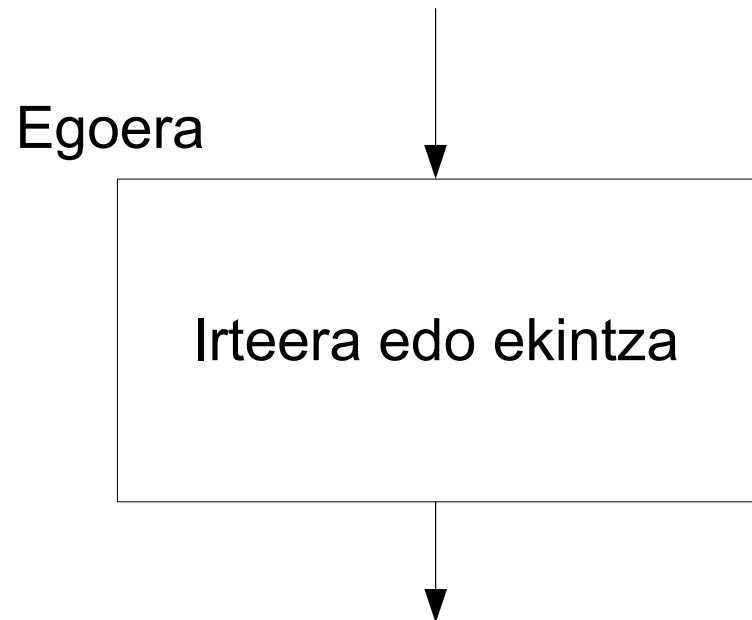
- Kontrol unitatea bi motatakoa izan daiteke:
  - Programagarria
  - Ez programagarria
- Kontrol unitate programagarriak memoria batetik aginduak jaso ditzake, beraz programa bat betetzeko gaitasuna dauka → Ordenagailua
- Kontrol unitate ez programagarria funtzio bakar bat betetzeko diseinatua dago → Ez dago programarik

# ASM diagrama

- Oso diseinu konplexuak izan daitezke zirkuitu sekuentzial sinkronoak
- Sarrera eta irteera ugari dauzkaten diseinu konplexuak, taula eta egoera diagramaren bidezko diseinua oso zaila da
- Diseinu sekuentzial konplexuak askoz errezago egin daitezke ASM (egoera makina algoritmikoa) diagramen bidez

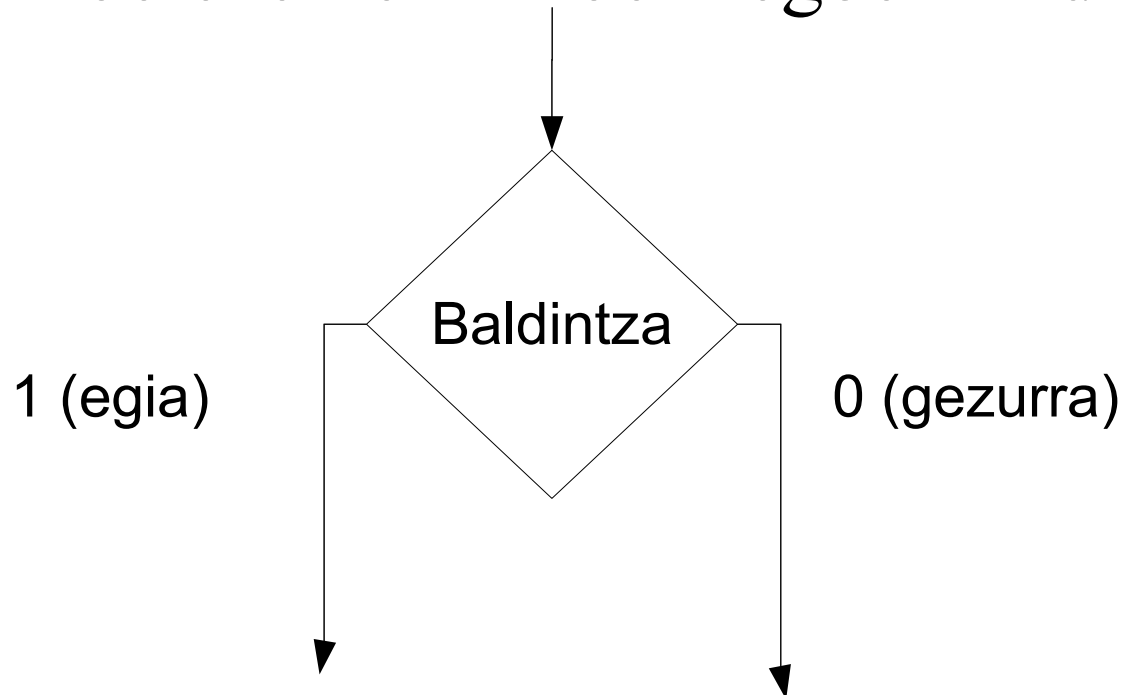
# ASM diagrama

- Egoera: Egoera eta hori dagokien irteera balioak
- Bakarrik lean jartzen den irteeraren izena idazten dugu
- Ekintza sinpleak ere adiera daitezke



# ASM diagrama

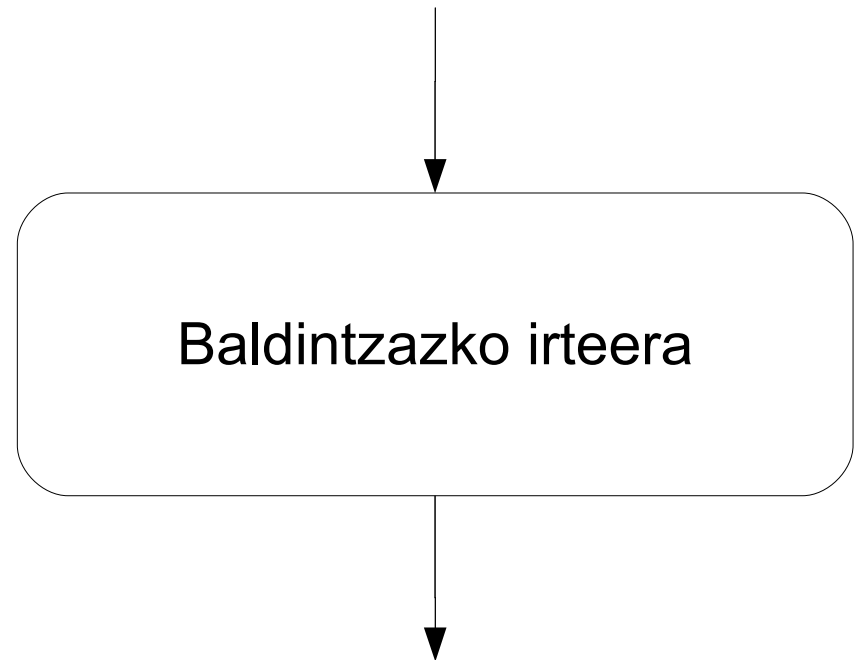
- Aukera: Trantsizioak definitzen ditu
- Baldintza zirkuitu sekuentzialaren sarrera baten arabera izango da
- Baldintza betezearen arabera egoera aldatuko da





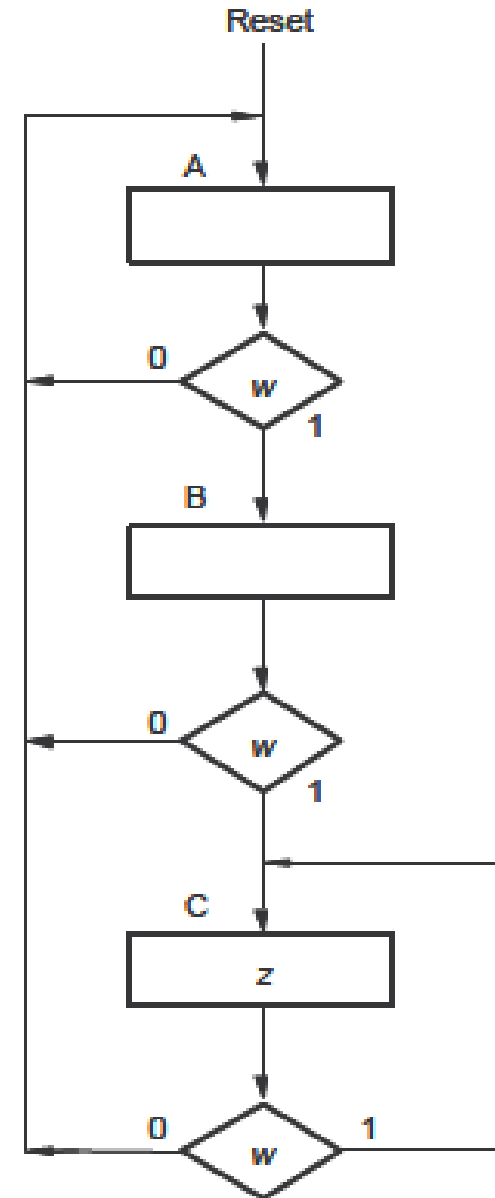
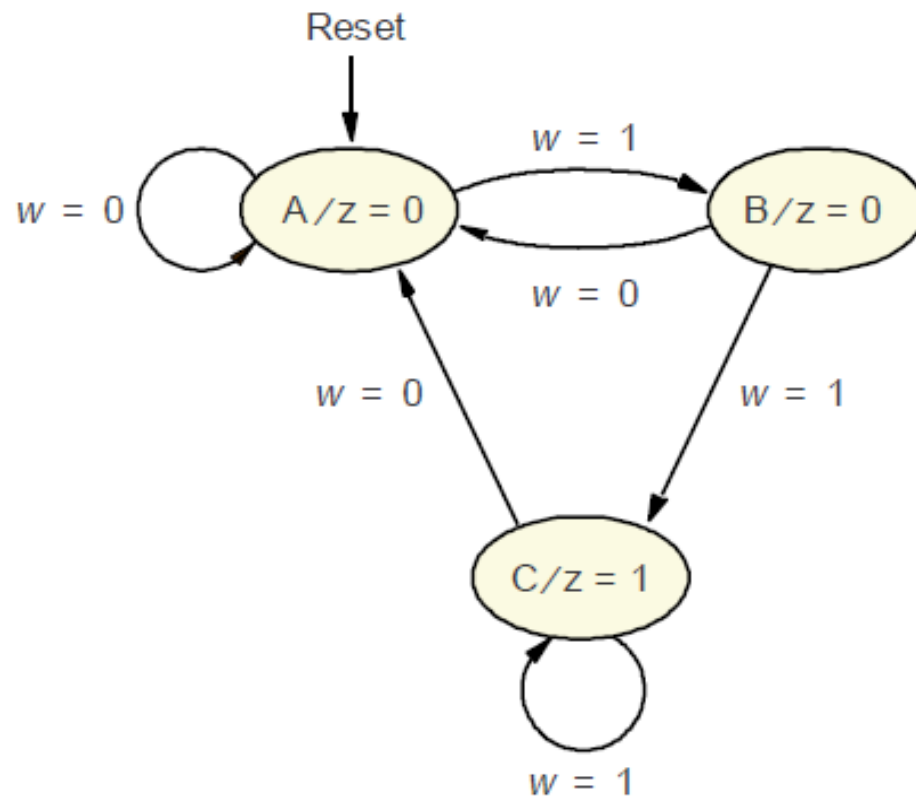
# ASM diagrama

- Baldintzazko irteera:  
Irteerako balio bat  
definitzen du
- Aukera baten  
emaitzaren arabera  
izango da  
baldintzazko  
irteeraren balioa



# ASM diagrama

- ASM diagramak zirkuitu sekuentzial baten portaera irudikatzen du, egoera diagramak egiten duen bezala:



# ASM-ren bidezko diseinua

- Diseinua egiteko, zirkuituaren portaeraren deskripzio batetik hasita, pausu hauek bete egin behar ditugu:
  - Ekintza zerrenda edo *algoritmo* bat osatuko dugu
  - Algoritmoari jarraituz, *datu bidearen bloke diagrama* diseinatuko dugu
  - Datu bide eta kontrol unitatearen arteko *konexio diagrama* bat egingo dugu, kontrol seinaleak ondo zehaztuz
  - Kontrol unitatearen *ASM diagrama* zehatz bat egingo dugu, seinale guztiak sartuta, diseinuari e la unidad de control, con todas las señales

# 1-eko kontagailua

- Erregistro baten dagoen datu baten 1-eko kopurua kontatzen duen zirkuitu bat diseinatu
- Algoritmoa:

$B=0$

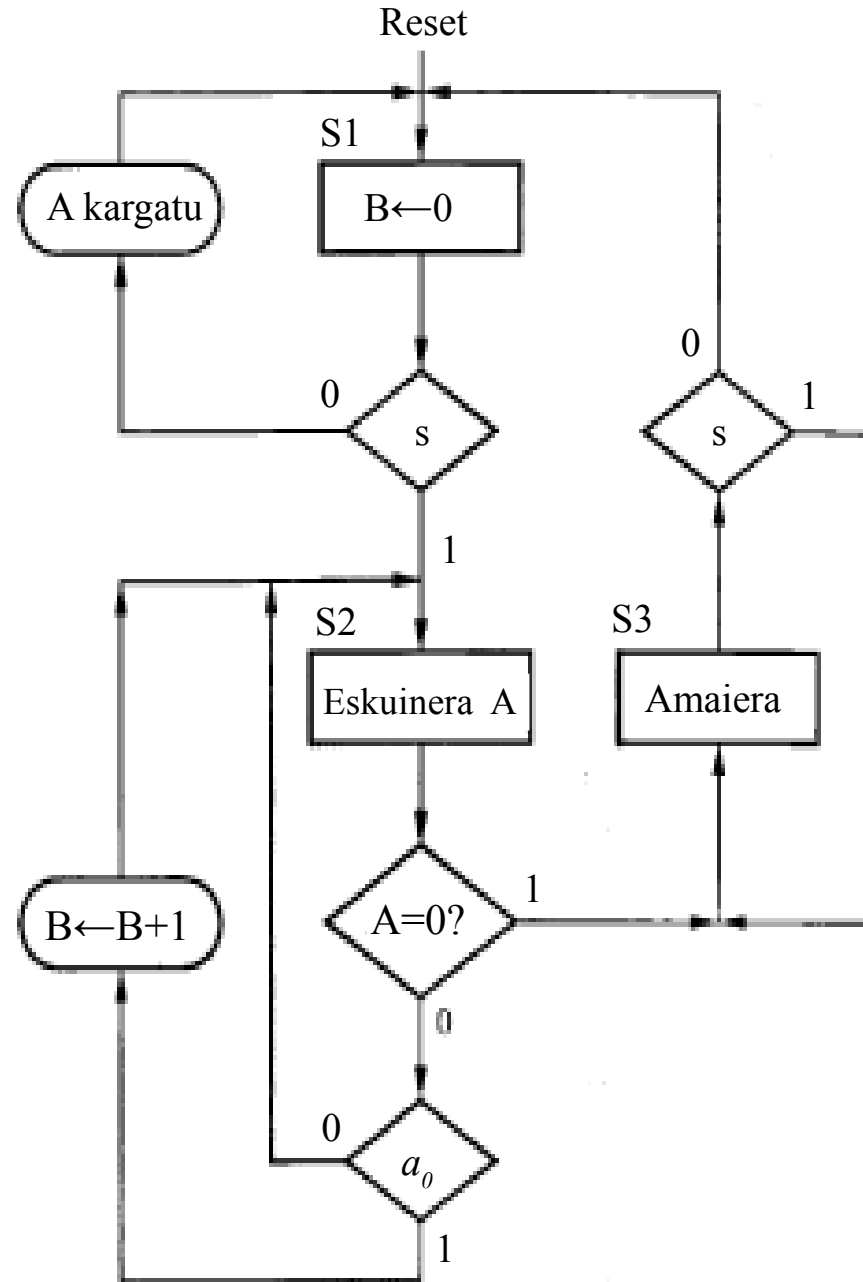
Errepikatu  $A \neq 0$  den bitartean:

$A_0=1$  denean,  $B=B+1$

$A \leftarrow$  Eskuinera desplazatu  $A$

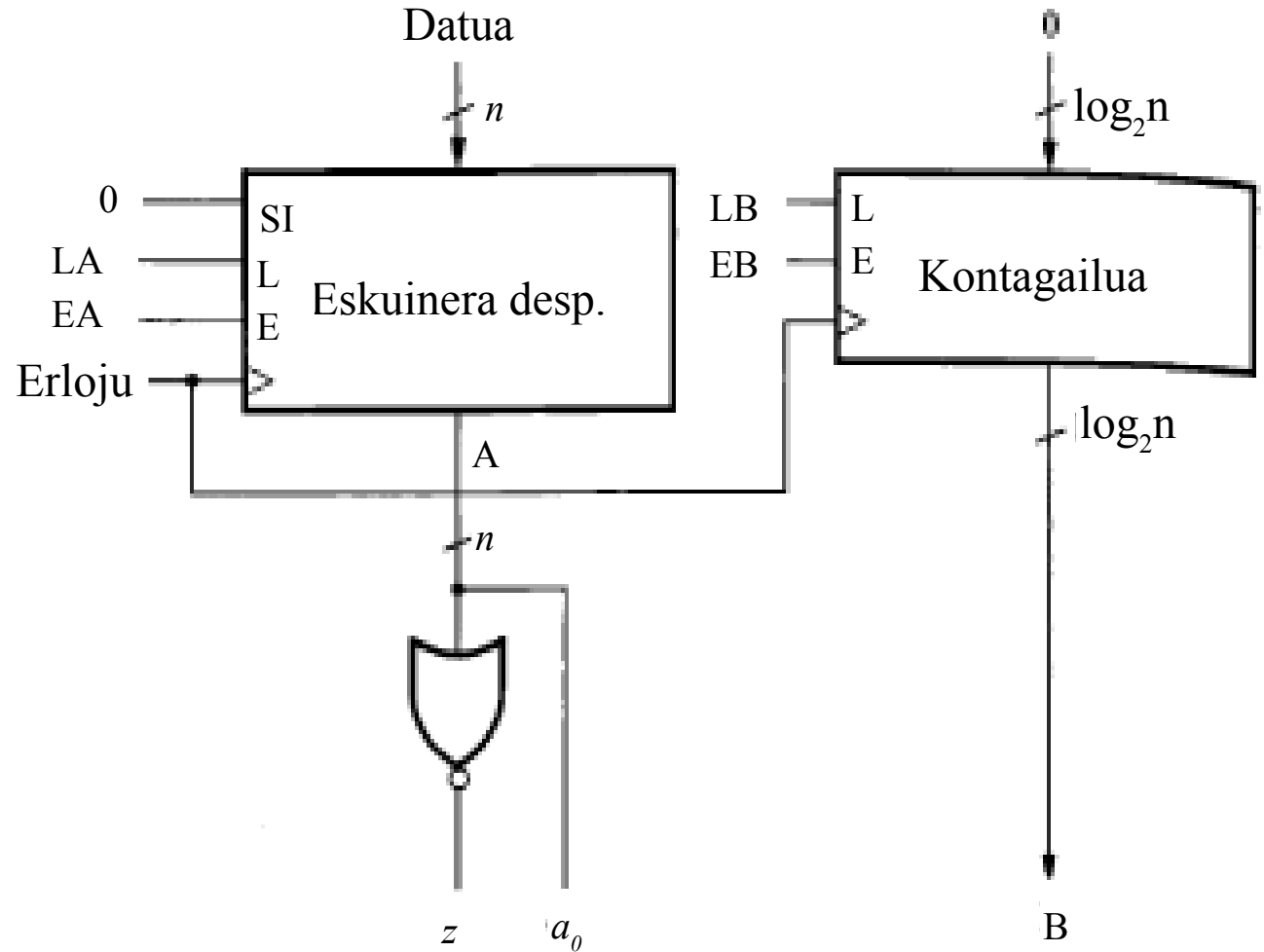
# 1-eko kontagailua

- Garatu algoritmoarekin batera doan lehenengo ASM bat osatu dugu (zehazpena ez da osoa)



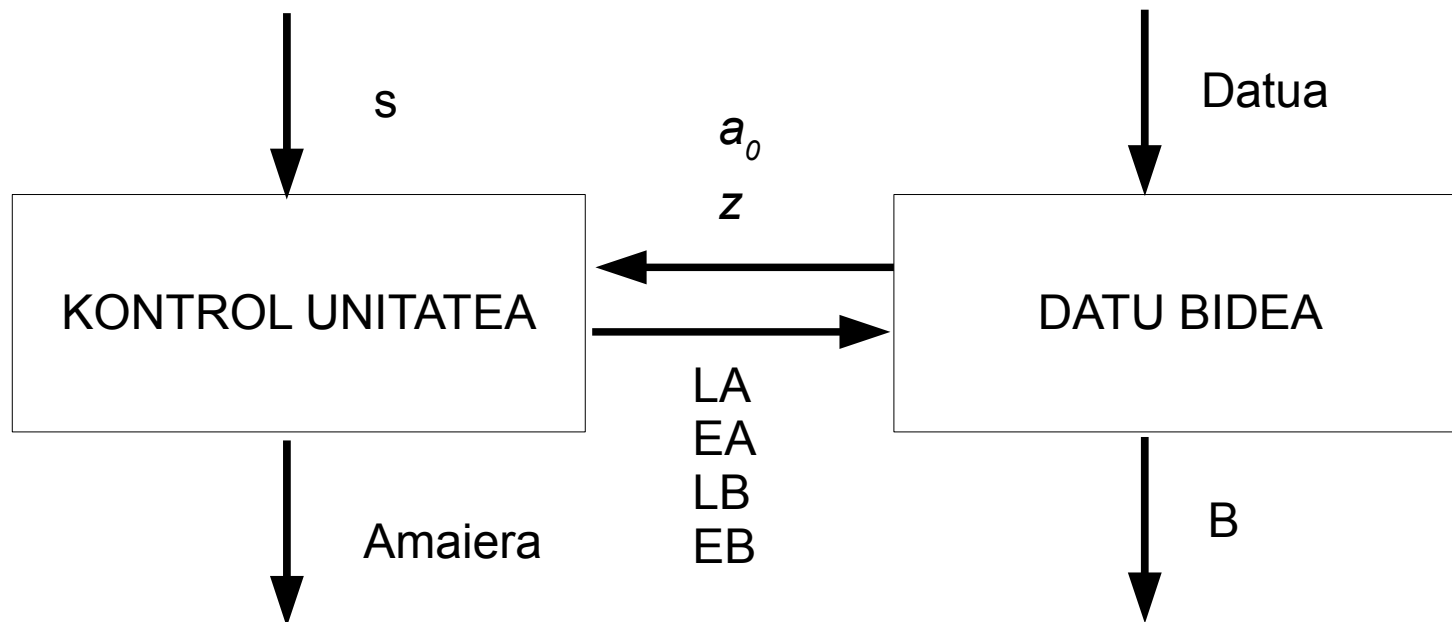
# 1-eko kontagailua

Algoritmoa gauzatzeko, kontagailua, eskuinerako desplazamendu ko erregistroa eta NOR ate bat (jakiteko ea  $A \neq 0$  den) behar ditugu



# 1-eko kontagailua

- Kontrol unitatea eta datu biden arteko konexio diagrama:



# 1-eko kontagailua

- Kontrol unitatearen diseinu osoa garatzen dugu, seinale guztiak barne
- ASM diagrama honek adierazten duen zirkuitu sekuentzialak behar ditugu kontrol seinale guztiak osatzen ditu

