

Latvijas Universitāte
Datorikas fakultāte

CPU. DataPath (tuprinājums) zarošanās

Kurss "ievads digitālajā projektēšanā"
Lekcija 07.10.2011

Autors: Rinalds Ruskuls

Jautājumi Jums...

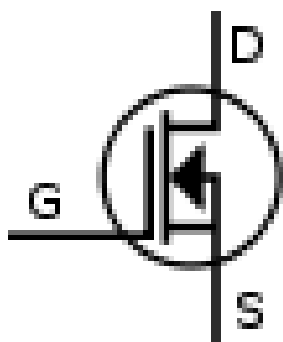
- Kā veicas ar mājas darbiem?
- Kā veicas ar praktiskajiem darbiem?
Kādas līdz šim izklāstītās tēmas būtu jāatkārto vai jāpapildina?
- Pirmā kursa darba uzdevums?

MD1 analīze

- Ir iesūtīti 8 mājas darbi
- Tipiskās kļūdas:
 - Nepareizs tranzistoru lietojums
 - Nav izpildīts līdz galam

Tranzistoru lietojums

- Tranzistors ir apgriezts – S un D ir apmainīti vietām



NMOS tranzistors strādās, ja:

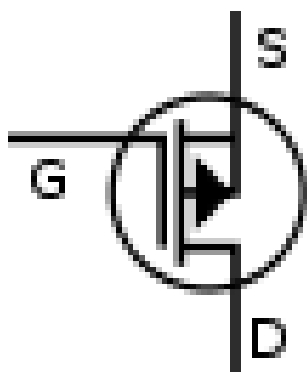
- 1) Uds spriegums ir pozitīvs

NMOS tranzistors nestrādās, ja:

- 1) Uds spriegums ir negatīvs

Tranzistoru lietojums

- Tranzistors ir apgriezts – S un D ir apmainīti vietām



PMOS tranzistors strādās, ja:

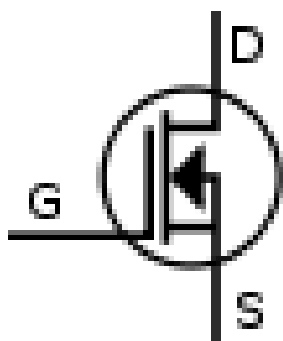
1) Uds spriegums ir negatīvs

PMOS tranzistors nestrādās, ja:

1) Uds spriegums ir pozitīvs

Tranzistoru lietojums

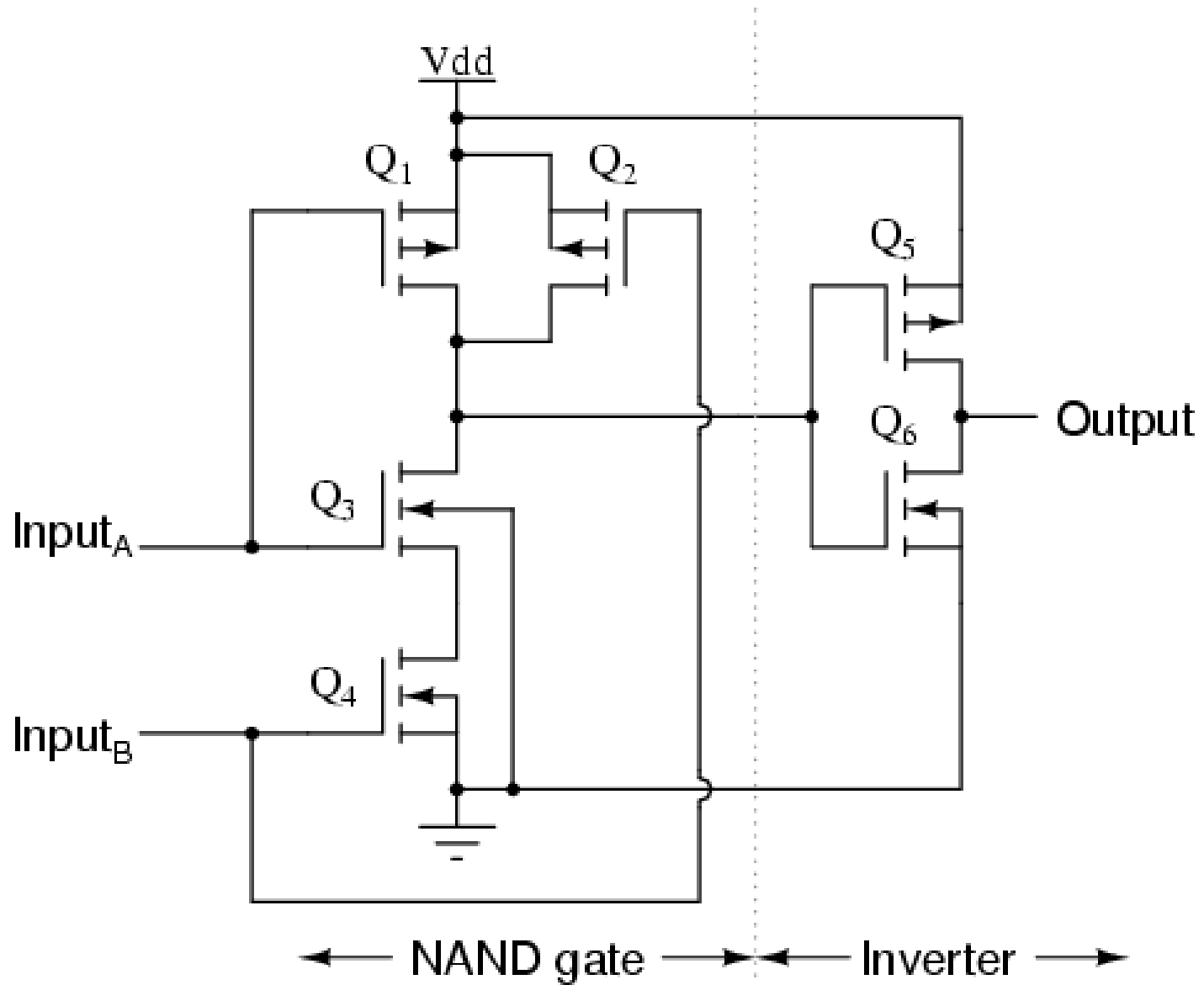
- Tranzistors nav agriezts – vadības spriegums nav stabils



Ja sprieguma līmenis starp U_{ds} mainās, tad

- 1) Mainās arī U_{gs} sprieguma vērtības
- 2) tranzistors atvērsies pie dažādām U_{gs} vērtībām
- 3) tranzistora darbība var kļūt sarežģīti paredzama

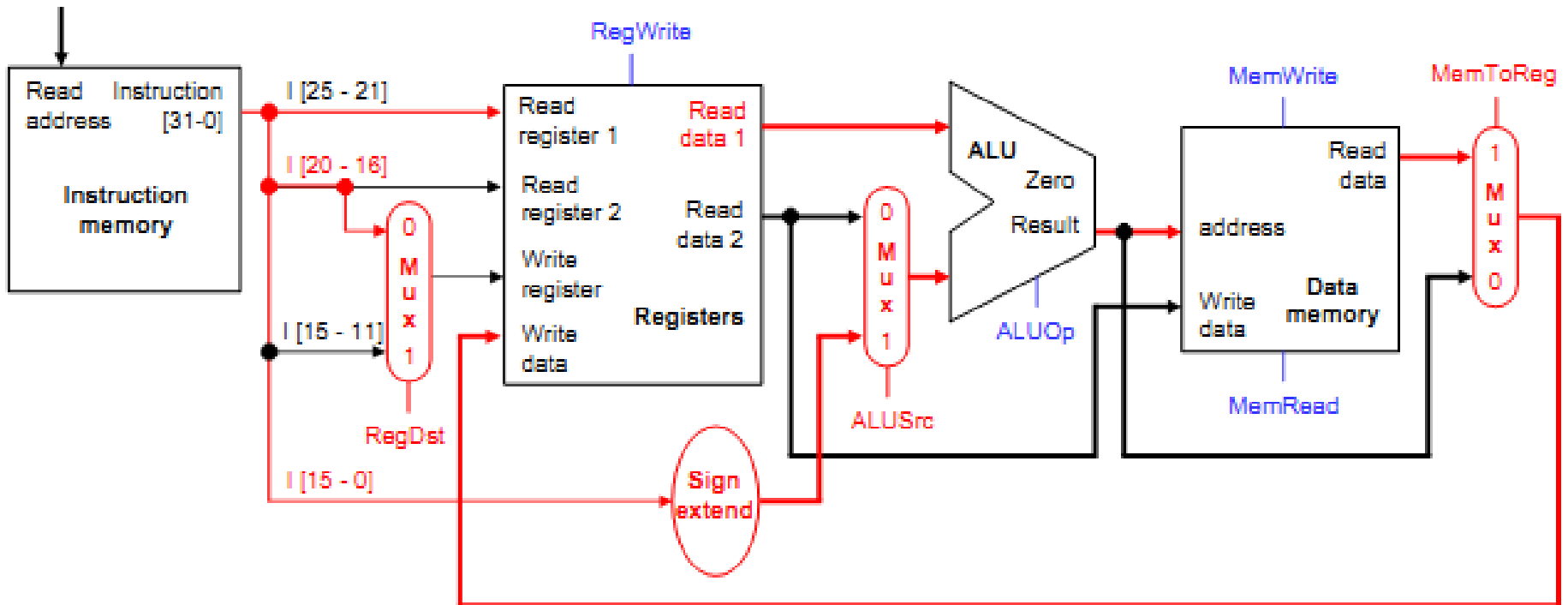
CMOS AND gate



Vientakts CPU implementācija

- Pagājušajā lekcijā tika izstāstīta vienkārša *MIPS CPU* arhitektūras implementācija, kas atbalsta šādas operācijas:
 - Aritmētiskās
 - Datu ierakstīšana / nolasīšana
- Pagājušajā lekcijā tika apskatīts vientakts MIPS CPU
 - Katra operācija izpildās vienā taktī
 - Uzsvars – datu maģistrāle

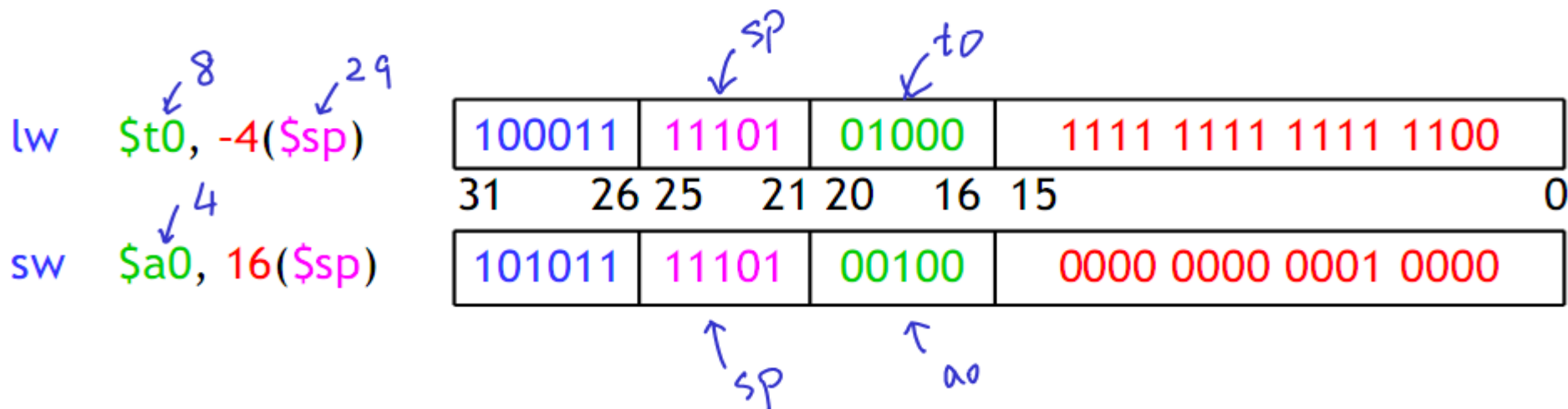
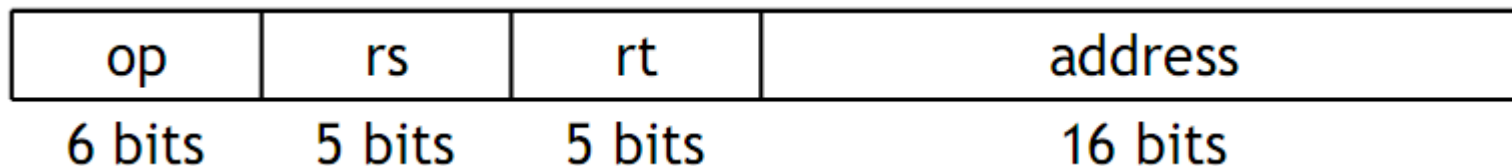
Shēmas kopskats



/ komandu izpildīšana

Komandas kā lw, sw, beq izmanto I tipa kodējumu

- ▶ rt ir izejas reģistrs prieks lw, bet priekš sw, beq ieejas
- ▶ address – 16 bitu konstante

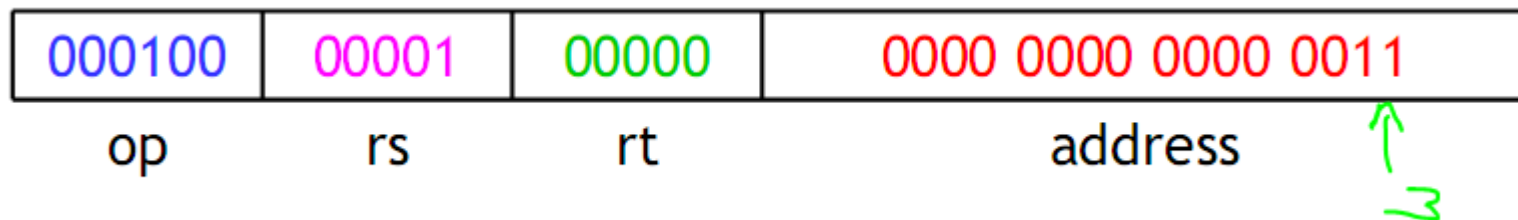


Beq komandas pielietojums

- **Beq** komandas gadījumā konstante nav adrese, bet gan instrukcijas atmiņas nobīde

```
beq  $at, $0, L
add  $v1, $v0, $0
add  $v1, $v1, $v1
j    Somewhere
L:   add  $v1, $v0, $v0
```

Katra instrukcija ir 4 baitus gara, līdz ar to reālā instrukcijas atmiņas nobīde tiek veikta par 12 baitiem ($3 \times 4 = 12$)

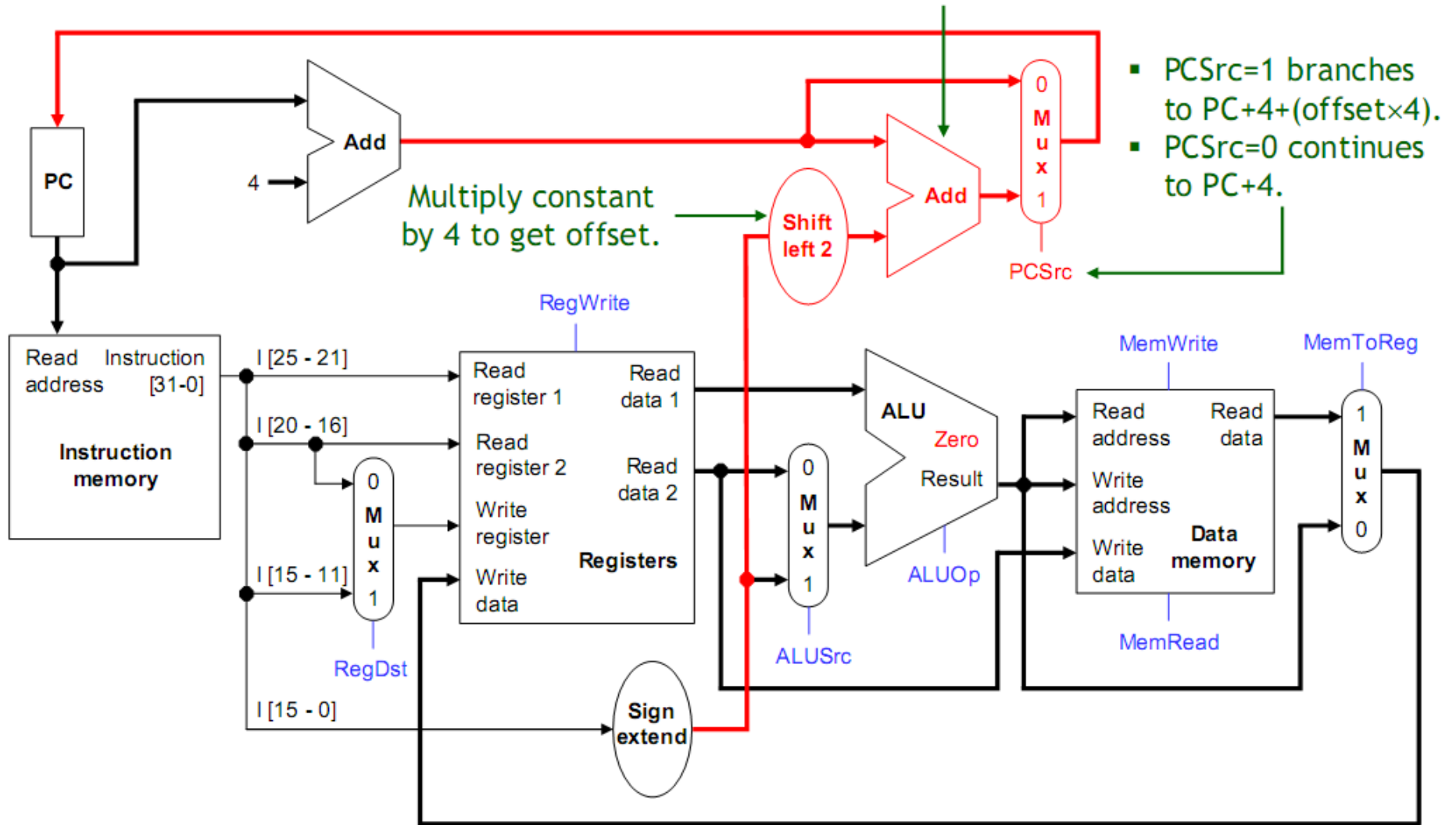


Kā izpilda **beq** komandas

- Nolasa no atmiņas instrukciju
- Nolasa attiecīgo reģistru vērtības
- ALU atņem nolasītās reģistru vērtības, ja
 - rezultāts ir nulle, izpilda instrukcijas nobīdi par uzdoto vērtību – **$PC + 4 + (offset \times 4)$**
 - nav – inkrementē instrukcijas adresi par konstantu vērtību – **$PC + 4$**

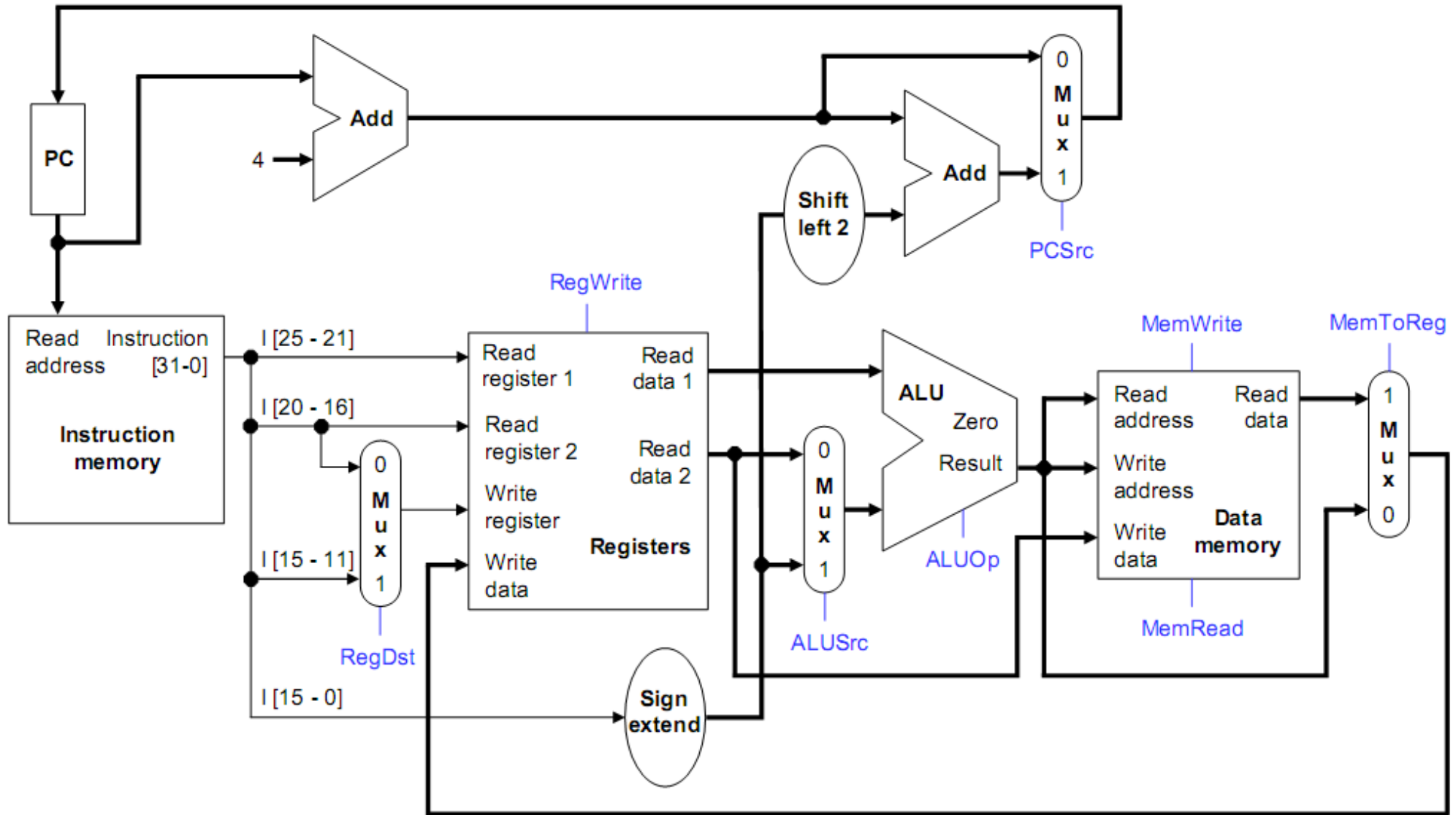
Branching hardware

We need a second adder, since the ALU is already doing subtraction for the beq.



CPU. Datapath. ALU

Gala variants



CPU. Datapath. ALU

Kontroles bloka tabula

Operation	RegDst	RegWrite	ALUSrc	ALUOp	MemWrite	MemRead	MemToReg
add	1	1	0	010	0	0	0
sub	1	1	0	110	0	0	0
and	1	1	0	000	0	0	0
or	1	1	0	001	0	0	0
slt	1	1	0	111	0	0	0
lw	0	1	1	010	0	1	1
sw	X	0	1	010	1	0	X
beq	X	0	0	110	0	0	X

ALUOp – R tipa instrukcijām, atkarīgas no instrukciju FUNC sadaļas

Lw un sw – vienīgās operācijas, kas izmanto konstantes apgabalu, tās izmanto ALU, lai izrēķinātu efektīvo atmiņas vērtību

Sw un beq – komandas, kas neraksta reģistros

CPU veikspēja

- Vientakts CPU veikspēju nosaka maksimāli lielākā aizture, lai veiktu operāciju
- Piemēram, ja CPU blokiem ir šādas aiztures
 - Instrukciju atmiņai – 200 ps;
 - Reģistru failam – 200 ps (lasīšana/rakstīšana)
 - ALU – 100 ps
 - Datu atmiņai – 400 ps
- 900 ps $\Rightarrow F = 1.11\text{GHz}$

Idejas, kā varētu paātrināt CPU veikspēju?

Kursadarbi

- Pirmais kursadarbs
 - Indivīduāls darbs
 - Shēmtēhniski Izveidot **32 bitu ALU**
- Otrais kursadarbs
 - Grupās pa 2
 - Vienkārša, pabeigta ciparu shēma (**running lights?**)
 - HDL valodas un/vai shēmtēhniski
- Trešais kursa darbs
 - Grupās pa....?
 - Sarežģītāka ciparu shēma – pa vairākiem cilvēkiem grupā
 - HDL valodas un/vai shēmtēhniski

Ieteikumi pirmajam kursa darbam

- Shēmas plānošana
- Secīga funkciju izveide
 - Loģiskās funkcijas
 - AND, OR, XOR, NOT
 - Aritmētiskās funkcijas
 - Sakaitīšana, atņemšana
 - Salīdzināšanas
 - Lielāks, mazāks, vienāds
- Funkciju apvienošana