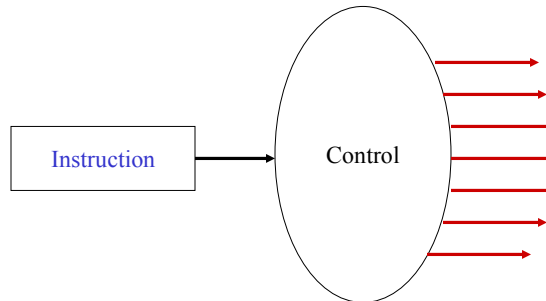


1/31/2007

CSE378 Control unit
Single cycle impl.

1

Control As a Function



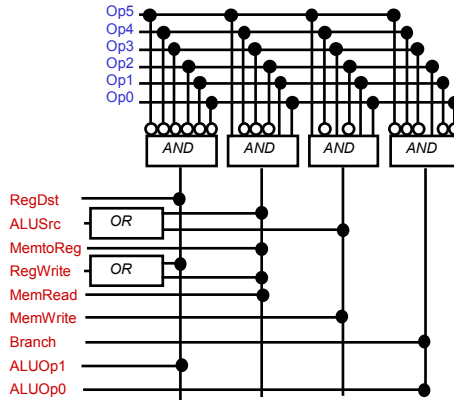
1/31/2007

CSE378 Control unit
Single cycle impl.

2

Computing a Function

Signal	R-type	w	sw	beq
Op5	0	1	1	0
Op4	0	0	0	0
Op3	0	0	1	0
Op2	0	0	0	1
Op1	0	1	1	0
Op0	0	1	1	0
RegDest	1	0	X	X
ALUSrc	0	1	1	0
MemtoReg	0	1	X	X
RegWrite	1	1	0	0
MemRead	0	1	0	0
MemWrite	0	0	1	0
Branch	0	0	0	1
ALUOp1	1	0	0	0
ALUOp0	0	0	0	1

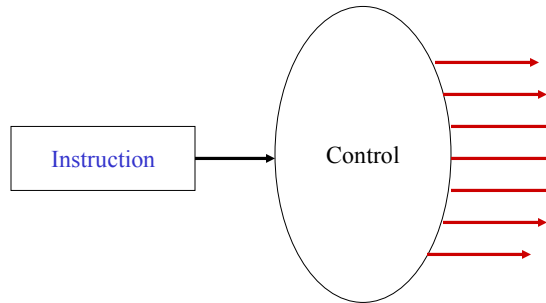


1/31/2007

CSE378 Control unit
Single cycle impl.

3

Are We Done?



1/31/2007

CSE378 Control unit
Single cycle impl.

4

Problems

- Function is “too big” if idea is applied directly
 - 2^{32} distinct inputs!
 - 32-way and gates
 - Want to reduce the “size” of the function
- Some control lines are not just a function of the instruction
 - Example: control of PCSrc mux depends on whether branch is taken or not
 - Some control functionality must be implemented as logic in the data path

1/31/2007

CSE378 Control unit
Single cycle impl.

5

Reducing Function Size: Part 1

- Instead of computing control signals at instruction execution time, compute them at assembly time
 - I.e., define the instruction encoding to include control bits
 - Simply copy the bits from the instruction encoding
- The (unnaturally trivial) example for MIPS:
 - Register numbers

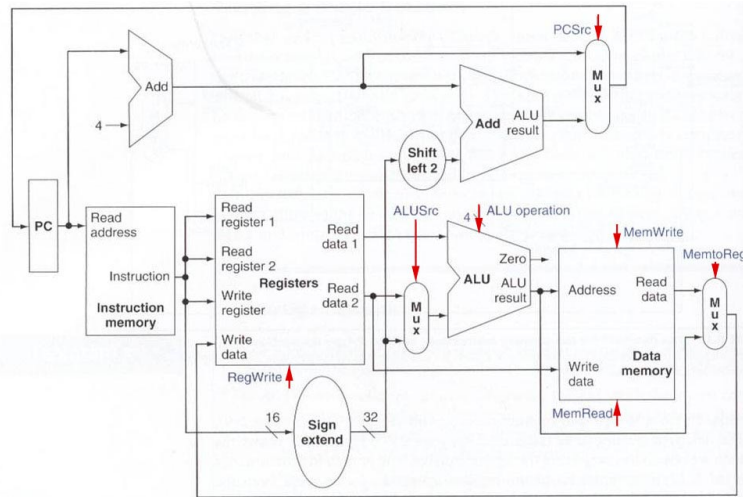
SPECIAL 000000	rs	rt	rd	0 00000	ADD 100000
-------------------	----	----	----	------------	---------------

1/31/2007

CSE378 Control unit
Single cycle impl.

6

With Some Control Encoded in Instructions



1/31/2007

CSE378 Control unit
Single cycle impl.

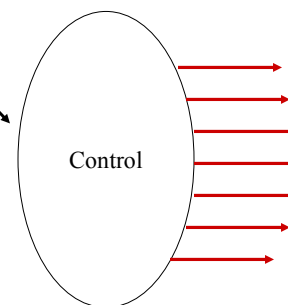
7

Modified Control Function



$$2^{(6+6)} = 4096 \text{ inputs}$$

$$\text{(vs. } 2^{32} = 4\text{B inputs)}$$



1/31/2007

CSE378 Control unit
Single cycle impl.

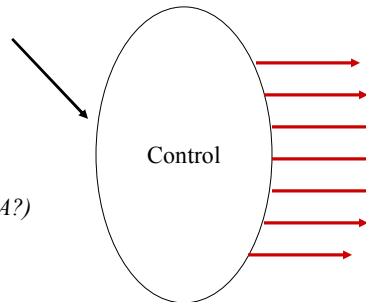
8

Can We Reduce Size Even More?

LW 100011	base	rt	offset		
ADDI 001000	rs	rt	immediate		
SPECIAL 000000	rs	rt	rd	0 00000	ADD 100000

$$2^{(6+6)} = 4096 \text{ inputs}$$

(Just use "don't cares" in PLA?)



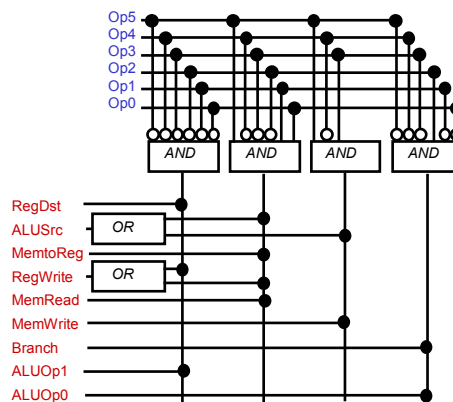
1/31/2007

CSE378 Control unit
Single cycle impl.

9

Don't Cares on Input

Signal	R-type	lw	sw	beq
Op5	0	1	1	0
Op4	0	0	0	0
Op3	0	0	1	0
Op2	0	0	X	1
Op1	0	1	X	0
Op0	0	1	X	0
RegDest	1	0	X	X
ALUSrc	0	1	1	0
MemtoReg	0	1	X	X
RegWrite	1	1	0	0
MemRead	0	1	0	0
MemWrite	0	0	1	0
Branch	0	0	0	1
ALUOp1	1	0	0	0
ALUOp0	0	0	0	1



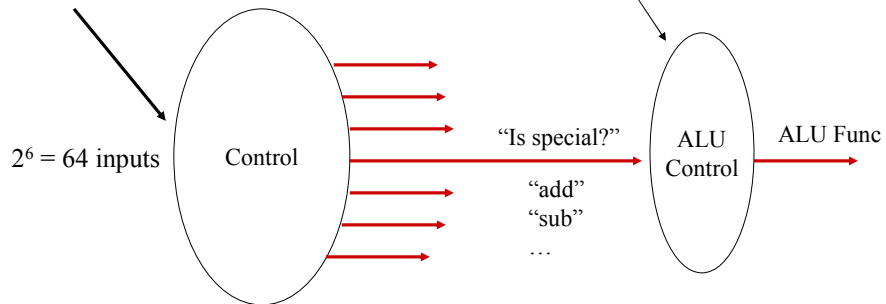
1/31/2007

CSE378 Control unit
Single cycle impl.

10

Can We Reduce Size Even More?

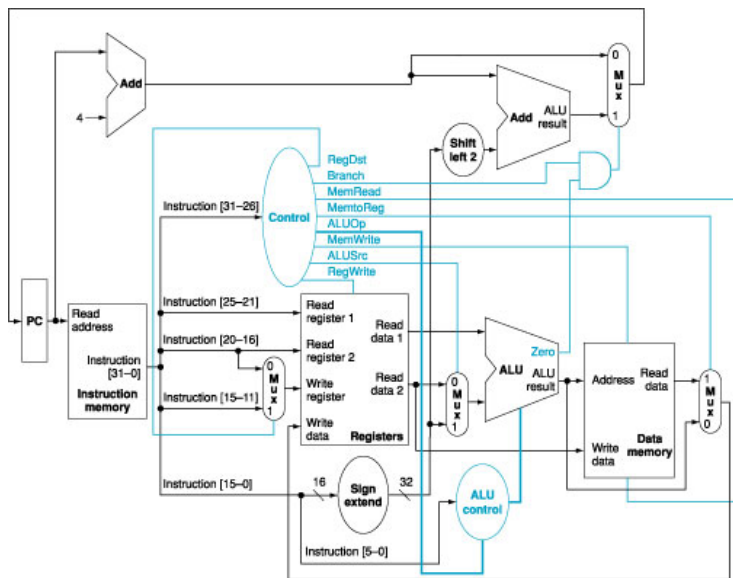
LW 100011	base	rt	offset		
ADDI 001000	rs	rt	immediate		
SPECIAL 000000	rs	rt	rd	0 00000	ADD 100000



1/31/2007

CSE378 Control unit
Single cycle impl.

11

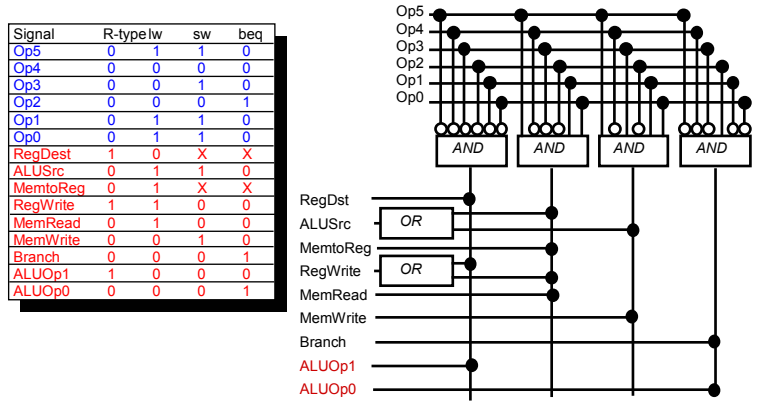


1/31/2007

CSE378 Control unit
Single cycle impl.

12

The Main Control



1/31/2007

CSE378 Control unit
Single cycle impl.

13

ALU Control

ALUOp1	ALUOp0	F5	F4	F3	F2	F1	F0	OP
0	0	X	X	X	X	X	X	0010
X	1	X	X	X	X	X	X	0110
1	X	X	X	0	0	0	0	0010
1	X	X	X	0	0	1	0	0110
1	X	X	X	0	1	0	0	0000
1	X	X	X	0	1	0	1	0001
1	X	X	X	1	0	1	0	0111

1/31/2007

CSE378 Control unit
Single cycle impl.

14

Can We Do Better?

- Why not just encode the ALU function bits directly in the func field of the R-type instructions?