

Outline

- Introduction
- Architecture of a RISC Processor**
- Implementation



Pirouz Bazargan Sabet

December 2008

Architecture

- Software visible registers**
- Memory Addressing
- The instruction set
- The exception / reset mechanism



Pirouz Bazargan Sabet

December 2008

Architecture

Software visible registers

Registers that can be manipulated (written and read) in the assembly language



Pirouz Bazargan Sabet

December 2008

Software Visible Registers

- 32 common 32-bit registers
Integer Registers

$R_0 \dots R_{31}$

Addressed from their number



Pirouz Bazargan Sabet

December 2008

Software Visible Registers

- R_0 : **The Trash Register**
A value written into R_0 is lost
 R_0 contains always 0
- R_{31} : **The Link Register**
When a subroutine is called, the
return address is saved into R_{31}



Pirouz Bazargan Sabet

December 2008

Software Visible Registers

- Two 32-bit registers : **HI** and **LO**
Used by multiply and divide instructions
- | | | |
|----------|----|---------------------------|
| Multiply | HI | 32 most significant bits |
| | LO | 32 least significant bits |
| Divide | HI | Result |
| | LO | Remainder |



Pirouz Bazargan Sabet

December 2008

Software Visible Registers

- Four 32-bit special registers : Coprocessor Registers
Needed to implement an operating system
- | | |
|----------|--|
| SR | Status Register |
| Cause | Cause Register (cause of exceptions) |
| EPC | Exception Program Counter (return
address in case of exception) |
| BadVAddr | Bad Virtual Address Register (illegal
memory address) |



Pirouz Bazargan Sabet

December 2008

Architecture

- **Software visible registers**
- **Memory Addressing**
- The instruction set
- The exception / reset mechanism



Pirouz Bazargan Sabet

December 2008

Memory Addressing

- 32-bit address \Rightarrow 4 Gbytes of memory space
- Read / Write operations
- 3 types of data :
 - Byte
 - Half-word (2 bytes)
 - Word (4 bytes)



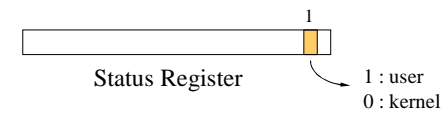
Pirouz Bazargan Sabet

December 2008

Memory Addressing

- The processor can operate under 2 modes
User / Kernel

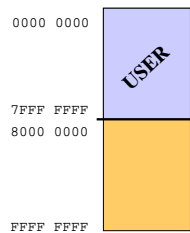
The current mode is defined by the Status Register



Pirouz Bazargan Sabet

December 2008

Memory Addressing



The memory space is divided into 2 parts

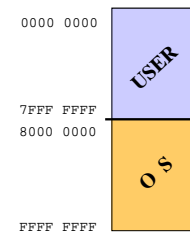
In *User* mode the processor can access only the addresses ranged from **0000 0000** to **7FFF FFF**



Pirouz Bazargan Sabet

December 2008

Memory Addressing



The OS protects the hardware against an error in a user program

This frontier protects the OS



Pirouz Bazargan Sabet

December 2008

