

Outline

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



Pirouz Bazargan Sabet

December 2008

Exception / Reset

Implementation of the :

- Reset mechanism
- Interrupt mechanism
- **Exception mechanism**



Pirouz Bazargan Sabet

December 2008

Implementation

- ❏ RISC vs. CISC
- ❏ Concept of pipeline
- ❏ An implementation of Mips
- ❏ **Pipeline's problems**



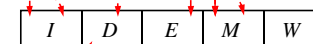
Pirouz Bazargan Sabet

December 2008

Exception

How stop the pipeline ?

- Overflow
- Illegal read address
- Illegal write address
- Coprocessor unusable
- Unknown instruction
- Syscall
- Break
- Trap
- Data bus error
- Instruction bus error
- Machine check

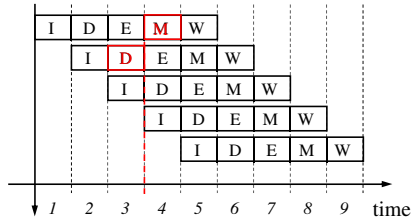


Pirouz Bazargan Sabet

December 2008

Exception

How stop the pipeline ?



Exceptions should be seen in the same order as the instructions (exact exception)



Pirouz Bazargan Sabet

December 2008

Exception

When stop the pipeline ?

Exceptions are events that denote a malfunction in the program

An exception is an error

⇒ the faulty instruction should NOT be executed

A faulty instruction should not modify the state of the system ... from the software point of view

Protect software visible registers and the memory

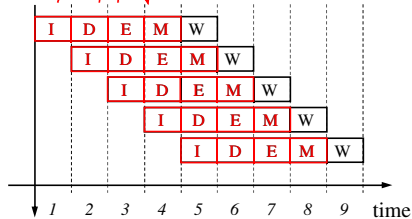


Pirouz Bazargan Sabet

December 2008

Exception

How stop the pipeline ?



Forward all the sources to the same cycle

Postpone the reaction to an exception

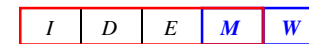


Pirouz Bazargan Sabet

December 2008

Exception

When stop the pipeline ?



Pirouz Bazargan Sabet

December 2008

Exception

How stop the pipeline ?

- Overflow
- Illegal read address
- Illegal write address
- Coprocessor unusable
- Unknown instruction
- Syscall
- Break
- Trap
- Data bus error
- Instruction bus error
- Machine check

Pirouz Bazargan Sabet

December 2008

Exception

When stop the pipeline ?

Pirouz Bazargan Sabet

December 2008

Exception

When stop the pipeline ?

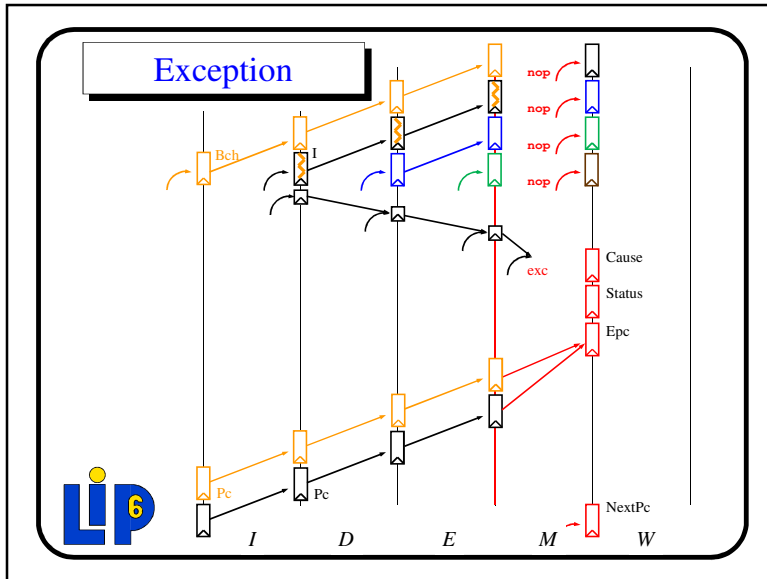
Pirouz Bazargan Sabet

December 2008

Exception

Pirouz Bazargan Sabet

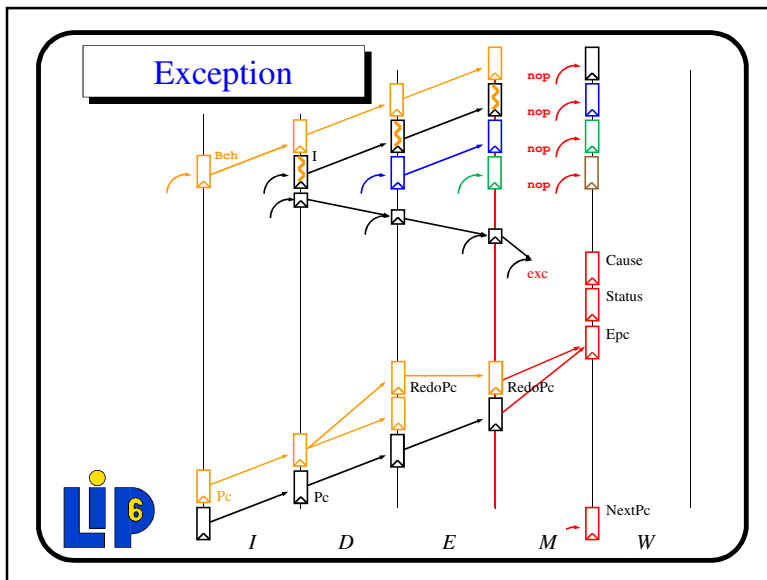
December 2008



Exception / Reset

Implementation of the :

- Reset mechanism
- **Interrupt mechanism**
- Exception mechanism



Interrupt

When stop the pipeline ?

Interrupts are events that require the processor to perform some operation

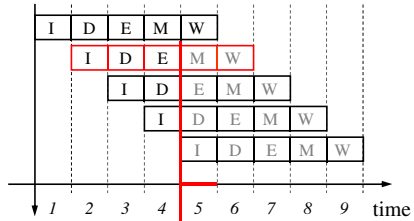
Interrupts are asynchronous
 ⇒ no emergency

Exceptions are events that denote a malfunction in the program

An exception is an error
 ⇒ the faulty instruction should NOT be executed

Interrupt

When stop the pipeline ?



Exception / Reset

Reset Abort the current program
Jump to the Reset Handler

- Initialize the address of the next instruction : $0xBFC0\ 0000$
- Initialize the Status Register : $0x0040\ 0004$
- Initialize the Cause Register : $0x0000\ 0000$
- Save the return address into Eepc
- Initialize the Exception Base Register : $0x8000\ 0000$



Exception / Reset

Implementation of the :

- **Reset mechanism**
- Interrupt mechanism
- Exception mechanism



Reset

