CS 350 Notes

Matthew Visser

Sep 15, 2011

1 Program Execution

- Registers
  - Has program counter, stack pointer, data for operations
- Memory
  - Stack frame: contains activation records, control flow, state of program.
    1. Local variables
    2. Return address
    3. Function parameters
  - Program code
  - program data
- CPU

2 MIPS Register Usage

See `kern/arch/mips/include/asmdefs.h` and slides `threads.pdf` page 2-3.

3 Threads

- Represents the control state of a program
- Has an associated *context* which consists of:

  - CPU state, including program counter, stack pointer, registers, execution mode (privilege)
  - a stack, which is located in address space of thread's process

- In short, contains all information you would need to suspend a program and then resume it later i.e. stack and cpu registers.

3.1 Advantages

- Don't have to wait for a program that will take a long time, can switch between that and doing something useful.
- Take advantage of multi CPU

3.2 Definitions

Thread Library: Code that manages threads.

Thread Control Block: Data structure that stores thread context.

3.3 Notes

- OS/161 stores register values on stack, just stores stack pointer.

3.4 Context Switching

Context Switching: The act of pausing one thread, and resuming or starting another.

What happens during this?

1. Save context
2. Decide what runs next (Scheduler)
3. Restore context of thread to run next *i.e. Dispatching* a new thread.

This is

- Architecture specific
- Thread must save/restore carefully, since context changes often
- Can be tricky to understand because it's hard to define what point a thread stops in, and when it resumes.

When the context switch happens, the dispatching function returns to the yield function of the *new* thread, not the old one.