Research Overview

Hardware/Software Codesign
of Real-time Systems

Dr. Morton

www.ece.uwaterloo.ca/~armorton/research

Embedded Systems

- Embedded System
  - cellphone, automobile, medical equipment
  - medium to large production volumes, upgrades
  - minimize size, power consumption, development time
- System on Chip (SoC)
  - CPUs, Memory, I/O and custom logic all on one IC

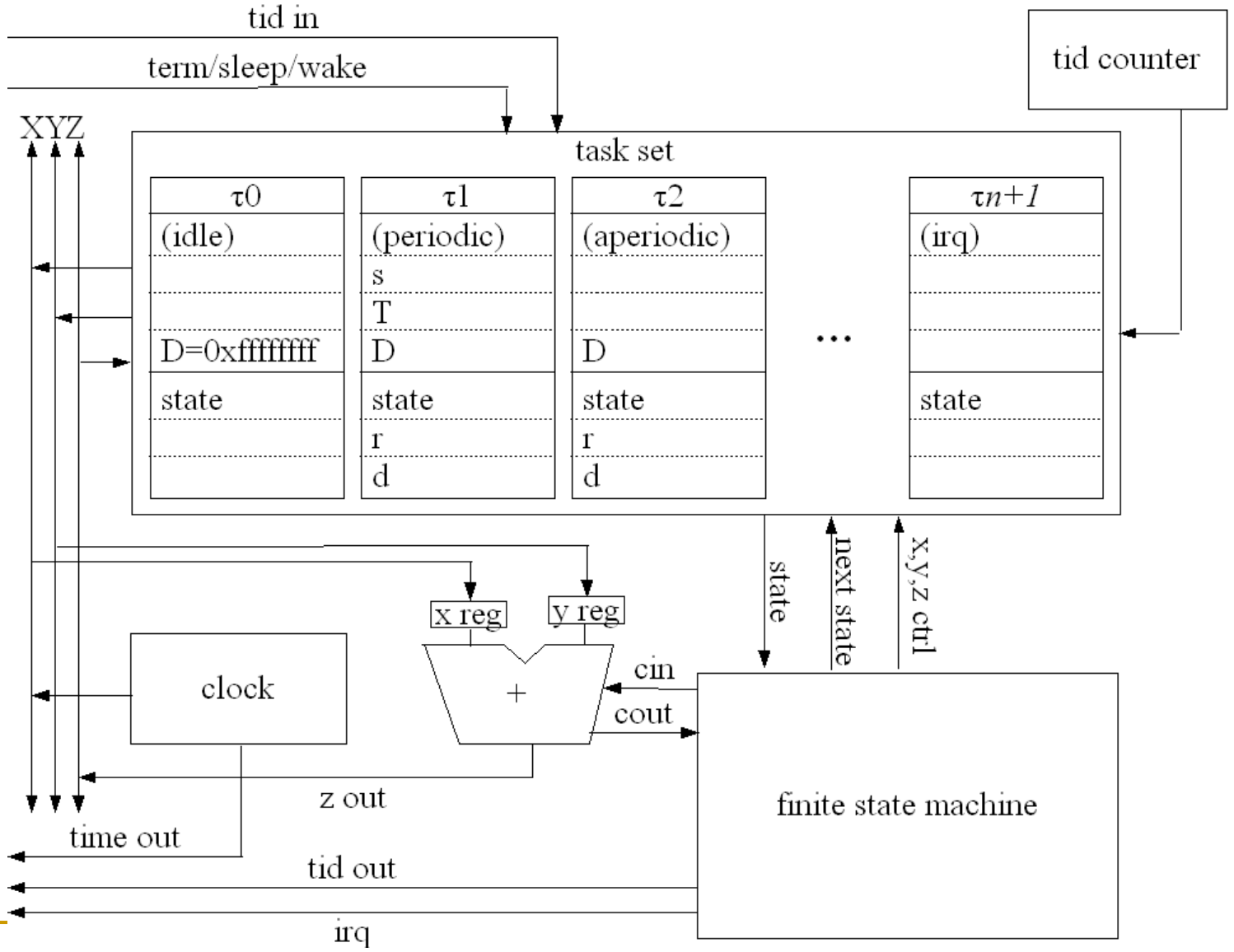
Hardware/Software Codesign

- Traditional approach: design hardware and hope that software can meet timing requirements
  - leads to overdesign
- Hw/Sw Codesign: design hw and sw in parallel
  - allow trade-offs at each design stage to help meet system-level objectives (area, power, timing)
- Question: How best to use resources to meet system timing requirements?

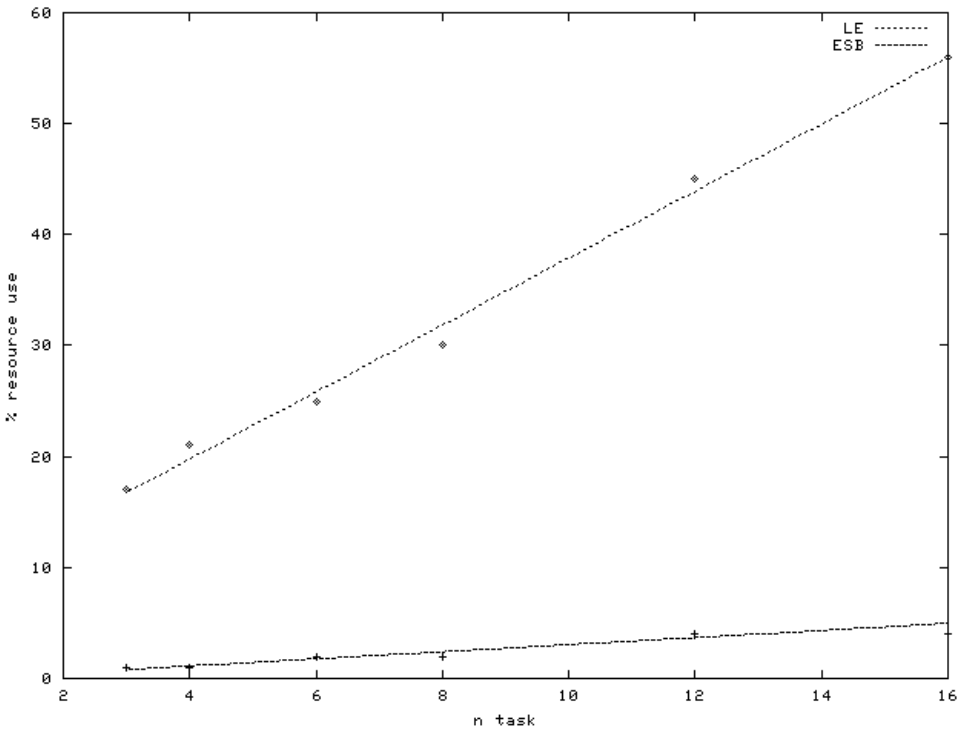
EDF Coscheduler

- Implement sw scheduler in hw
- EDF = Earliest Deadline First (preemptive)
  - task with earliest deadline executes
  - optimal scheduling policy
- Implementation
  - software: min-heap (log *n)*
  - *hardware: round-robin bidding*

Hardware Structure

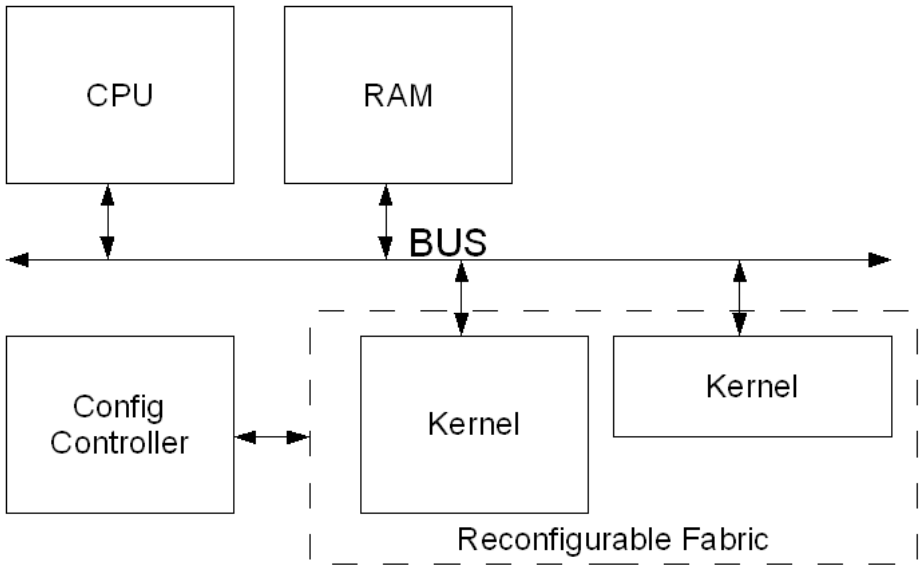


Results



- Case study
  - reduced kernel overhead by ~50%
  - fast event response
  - almost as much hw as a second CPU
  - benefits depend on application
- Commercial example
  - Sierra real-time kernel

Dynamically Reconfigurable Systems



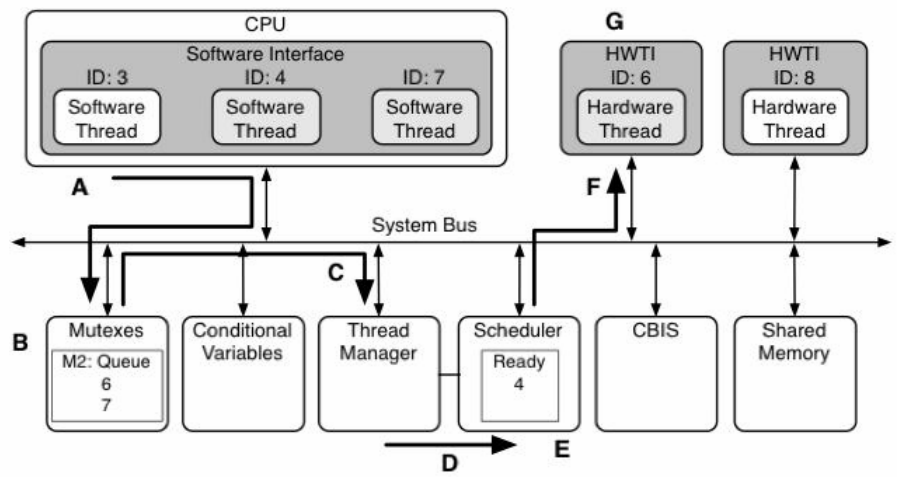

```
main(...) {  ...
    for(i=1 to 10) x(,,,);
    while(true) { ...
        y(...);}
...}
x(...) {
→           rand() ...}
y(...){
    for(j=...)
→   eliptic()
}
```

Configuration Scheduling

- Kernel configuration takes time
  - can outweigh time saved by using hw kernel in place of sw kernel
- Dynamic scheduler
  - chooses when to use hw kernel and when to use sw kernel
  - optimize performance by choosing when to configure and when not
  - can itself be in sw or hw

Hw/Sw Programming Model

- hThreads (David Andrews, U of Kansas)
- adapt posix threads to support high-level programming model of both software and hardware
  - sync and share data between hw and sw threads



http://wiki.ittc.ku.edu/hybridthread/images/c/cf/HthreadMutex.jpg

Hybrid Concurrency Framework

- pthreads is library-based concurrency
  - weaknesses: compiler, programmer error
  - program-based concurrency better
  - Java, Ada, uC++

- hybrid uC++ executive
- semaphore, monitor, active task