

Event based / Thread based programming for Wireless Sensor Networks

Uses:

Area Monitoring
Environmental
Monitoring
Greenhouse Monitoring
Aquatic Sensors
Agriculture



Main Constraints:

Limited Power
Harsh Environment
Node Failure
Dynamic N/W Topology
Unattended operation

Crossbow MicaZ Mote

4KB RAM, 12KB Flash Memory, 2AA Batteries, 3 LEDs, 2.25*1.25 by 0.25 inches

Event Based :

Run to completion
Not very efficient for large programs
Uses comparatively less memory
More power consumption
Fewer lines of code



Thread Based:

Interrupted by other threads
Can be modeled for large programs
Uses more memory because of stacks
Less power consumption
More lines of code

Mib510 Serial Programming board

Tiny OS (Written in NesC)

Limited Resources, Reactive con currency, Flexibility, Low Power

Programming Styles

```
module HelloWorldC {
  uses {
    interface Boot;
  }
  implementation {
    event void Boot.booted() {

    }

    dbg("RadioCountToLedsC", "Hello
    World.");
  }
}
```

```
module HelloWorldC {
  uses {
    interface Boot;
    interface Thread as TinyThread0;
  }
  implementation {
    event void Boot.booted() {
      call TinyThread0.start(NULL);
    }
    event void TinyThread0.run(void* arg) {
      for(;;) dbg("Boot", "Hello World "); }
  }
}
```

Using different measures like:

Lines of code, memory consumption, Power consumption, Complexity, Reliability

and using same programs with two different programming style to understand the tradeoffs inherent in two different programming styles and also which style of programming should be better in specific situation.