TinyOS Core WG Report

Philip Levis (speaking on behalf of the WG)

TTX4
4/28/07
Core WG Charter

• “The TinyOS 2.0 Core Working Group (core) decides what abstractions, services, and interfaces are in the TinyOS 2.0 core system, developing reference implementations when needed. While the core WG focuses on the hardware independent interfaces to hardware-specific abstractions (the basic OS services), it sometimes also addresses basic abstractions that many systems might use, such as data structures.”
Core Responsibilities

- TinyOS Service Interfaces
- Core
- Hardware

Hardware Independent Layers

- net2
- storage
- ...
Milestones

- July 7, 2006: TinyOS 2.0 beta2 release
- November 6, 2006: TinyOS 2.0 release
- March 20, 2007: All WG TEPs below 120 sent to SC
- April 29, 2007: TinyOS 2.0.1 release
- June/July 2007: TinyOS 2.0.2 release
TinyOS 2.0.1

- Low-power CC2420 stack (TEP 126)
- Refinement of resource/energy management (TEP 108)
- lib/printf
- sensorboards/mts300
- Additional tutorials
- Sample application: AntiTheft
- Bug fixes, optimizations, etc.
# Core WG TEPs

<table>
<thead>
<tr>
<th>Number</th>
<th>Title</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>TEP 1</td>
<td>TEP Structure and Keywords</td>
<td>Community Review</td>
</tr>
<tr>
<td>TEP 2</td>
<td>Hardware Abstraction Architecture</td>
<td>Sent to SC</td>
</tr>
<tr>
<td>TEP 101</td>
<td>Analog-to-Digital Converters</td>
<td>Finalized</td>
</tr>
<tr>
<td>TEP 102</td>
<td>Timers</td>
<td>Author Response</td>
</tr>
<tr>
<td>TEP 103</td>
<td>Permanent Data Storage (Flash)</td>
<td>Finalized</td>
</tr>
<tr>
<td>TEP 106</td>
<td>Schedulers and Tasks</td>
<td>Finalized</td>
</tr>
<tr>
<td>TEP 107</td>
<td>TinyOS 2.x Boot Sequence</td>
<td>Finalized</td>
</tr>
<tr>
<td>TEP 108</td>
<td>Resource Arbitration</td>
<td>Finalized</td>
</tr>
<tr>
<td>TEP 109</td>
<td>Sensor Boards</td>
<td>Community Review</td>
</tr>
<tr>
<td>TEP 111</td>
<td>message_t</td>
<td>Finalized</td>
</tr>
<tr>
<td>TEP 112</td>
<td>Microcontroller Power Management</td>
<td>Community Review</td>
</tr>
<tr>
<td>TEP 113</td>
<td>Serial Communication</td>
<td>Sent to SC</td>
</tr>
<tr>
<td>TEP 114</td>
<td>SIDs: Source and Sink Independent Drivers</td>
<td>Community Review</td>
</tr>
<tr>
<td>TEP 115</td>
<td>Power Management of Non-Virtualized Devices</td>
<td>Sent to SC</td>
</tr>
<tr>
<td>TEP 116</td>
<td>Packet Protocols</td>
<td>Sent to SC</td>
</tr>
<tr>
<td>TEP 117</td>
<td>Pins and Buses</td>
<td>Author Response</td>
</tr>
</tbody>
</table>
Power Management

- A core advancement in TinyOS 2.0 is its integrated concurrency and power management.
- Using a hardware resource keeps it powered.
- Power trace from sample low-rate sensing application with no explicit power management:

![Power Trace Graph](image-url)
Roadmap

- Emergence of new platforms and chips
- Finalization of current TEPs
- Low-power CSMA radios
- Collaboration with net2 on link layer
- TinyOS 2.0.2: June 2007
Open Questions

• Release packaging?
  ✴ Responsibilities: core, or another WG?
  ✴ RPMs, tarballs, debian packages

• Toolchain?
  ✴ msp430-gcc
  ✴ avr-gcc

• Component libraries to simplify application development?
WG Members

7 Universities and 6 companies:

David Gay, Vlado Handziski, David Moss, Philip Levis, Kevin Klues, Jan Heinrich-Hauer, Andres Koepke, Phillip Huppertz, David Culler, Jonathan Hui, Gilman Tolle, Philip Buonadonna, Cory Sharp, Maxime Muller, Joe Polastre, Martin Turon, Benjamin Greenstein, Prabal Dutta, Lama Nachman, Jan Beutel, Robert Szewczyk, Martin Leopold and more...