Low-Power Wireless Sensor with SNAP and IEEE 1451 Protocol

Dr. Darold Wobschall
and
Sriharsha Mupparaju
Esensors Inc.
Goals

To describe ---

- Low-power wireless sensor requirements
- Advantages of compact SNAP protocol
- Need for IEEE 1451 smart transducer standard
- Wireless temperature sensor example
Network Sensor
Features and Applications

Features
- Automatic testing
- Plug and play
- Multiple sensors on one network or bus

Applications
- Machine to Machine (M2M) sensor data communications
- Wide area (Nationwide) data collection ability
Wireless Sensors
for short-range, unlicensed band

Grouped by Power Requirements
A. Significant power available
   line-powered or laptop size battery
B. Medium low power
   re-chargeable batteries or shorter life applications
C. Very low power
   Long operation (years) on small battery
Wireless sensors –
A. Significant power available

- Line-powered or laptop sized battery
- Uses transceiver
- Popular choice: WiFi (IEEE 802.11b), 2.4 GHz
  Components widely available (moderate cost)
  Good bandwidth
- Variation of TCP/IP protocol, mostly non-standard
Wireless sensors –
B. Medium low power

- Re-chargeable battery -- Uses transceiver

- Popular choices: Bluetooth (IEEE 802.15.1)
  - Low cost components (production scale)
  - Hard to interface to sensors on prototype scale
  - Moderate bandwidth

- Zigbee (IEEE 802.11.5)
  - Low bandwidth
  - Intermittent communication (sleep mode)
  - Star or Mesh Configurations

New device: Low-power WiFi (GainSpan)
Wireless sensors –
C. Very low power

- Coin size battery, non-rechargeable, lifetime of years
- Transmit only
- Popular choice: TI/Chipcon (433 MHz and 2.4 GHz)
  RF modules and microcontrollers available
  Low bandwidth
  Intermittent transmission (sleep mode)
Example – Wireless Serial Connection

- Wireless modules with RS232 I/O when connected to Dot 2 TIMS are similar to IEEE 1451.5 TIMs (wireless version of IEEE 1451).
- Data format and TEDS are the same (both follow the Dot 0 standard).

Dot 5 TIM built from a Dot 2 TIM and wireless transceiver
One of many sensor networks available.
Wireless Sensor Example
(low cost, long life, short messages)

- WS01 wireless temperature (and other) sensors
- 433 MHz FSK – range 10 to 100 meters, transmit only
- SNAP data protocol (header, data, crc)—8 to 12 bytes
- Dot 4 TEDS send periodically

Printed Circuit Board (back)  Photo (front)
Wireless Temperature Sensor Block Diagram
Wireless Temperature Sensor
Circuit Diagram
Wireless Sensor Receiver
Block Diagram

433 MHz

RF Transceiver \rightarrow \text{MicroComputer} \rightarrow \text{INTERNET Gateway} \rightarrow \text{INTERNET}

Receiver/ Transceiver
Circuit Boards for Wireless Receivers
SNAP Data Transmission Protocol

<table>
<thead>
<tr>
<th>Header</th>
<th>Address Field</th>
<th>Data Field</th>
<th>CRC field</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYNC (1-byte)</td>
<td>Header Composition (2 bytes)</td>
<td>Destination address (1 byte)</td>
<td>Source Address (3 bytes)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Sensor Data and TEDS (Dot4) sent using this format
Transmitted Wireless Data

Test Screen
(two transmissions shown, data in hexadecimal)
Received Data
(raw data and temperature in °C)

Three wireless sensors sending data

![Received Data Table]

<table>
<thead>
<tr>
<th>Received Data</th>
<th>ID</th>
<th>Reading</th>
</tr>
</thead>
<tbody>
<tr>
<td>pDET1031011</td>
<td>1</td>
<td>24.50</td>
</tr>
<tr>
<td>pDET202EA111</td>
<td>2</td>
<td>23.31</td>
</tr>
<tr>
<td>pDET3030DIII</td>
<td>3</td>
<td>24.41</td>
</tr>
</tbody>
</table>

Ready: pDET3030D
Opportunistic Mesh Network

- Wireless mesh network which minimized node-to-node transmission energy
- Includes error/missing data recovery
- Purpose: minimize battery power
- Uses receiver signal strength
Opportunistic Mesh Network Protocol
A review of the IEEE 1451 Smart Transducer Concept
IEEE 1451 Advantages

- Comprehensive enough to cover nearly all sensors and actuators in use today (not 20/80% approach)
- Many operating modes (buffered, no-buffer, grouped sensors, timestamps, timed data, streaming …)
- Extensive units, linearization and calibration options
- Multiple timing and data block size constraints handled.
- Compatible with most wired and wireless sensor buses and networks (point-to-point, mesh, TIM-to-TIM, mixed networks).
- Efficient binary protocol (especially suitable for wireless)
- Standard is 400+ pages for basic part, over 1500 page total
But the Complexity!

- A comprehensive standard is necessarily complex
- There was little adoption of the original IEEE 1451.2 (TII) standard because of its perceived complexity
- Manual preparation of the TEDS is not practical -- A TEDS compiler is needed
- A compliance test procedure is also desirable to prove that a design is correct

Munch – The scream
Main parts of IEEE 1451.0 (Dot 0) Standard

- Command /Response format
- Transducer Electronic Data Sheet (TEDS)
Dot 0 TEDS Format

- General format for each TEDS section:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Data Type</th>
<th>Number of Bytes</th>
</tr>
</thead>
<tbody>
<tr>
<td>-----------</td>
<td>----------------------</td>
<td>------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>-----------</td>
<td>----------------------</td>
<td>------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>TEDS Length</td>
<td>Data Block</td>
<td>Data Type</td>
<td>Number of Bytes</td>
</tr>
<tr>
<td>-------------</td>
<td>----------------------</td>
<td>------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>Data Block</td>
<td>Checksum</td>
<td>UInt16</td>
<td>2 bytes</td>
</tr>
<tr>
<td>-------------</td>
<td>----------------------</td>
<td>------------</td>
<td>-----------------</td>
</tr>
</tbody>
</table>

- Binary TEDS Tuple format for each data block:

  Type-Length-value (TLV)

  Example: 01 02 A3 04

  Field type is 1, Length is 2 bytes, field value is “A304” hex

- Field example: Meta-TEDS (TEDS # 1)

  13: Number of Implemented Transducer Channels (default=1)
Dot 0 TEDS Format

- **Required TEDS**  [Memory block with defined format]
  - MetaTEDS
  - Channel TEDS
  - Calibration TEDS (unless SI units)
  - Xdr-name TEDS
  - Phy TEDS
  - Also optional TEDS

- **Data Transmission**  [specific octet format]
  - TEDS/Status requests
  - Triggering and configuration
  - Sensor read commands and data return
  - Actuator write commands and data sending
Wireless (Dot 5) Options

- IEEE 1451.5 protocols are based on existing wireless protocols used for sensor networking (mostly additions to the OSI Application Layer)
- NCAP (gateway) Network (e.g. Internet) format uses Dot 0
- Current options are:
  - WiFi (IEEE 802.11)
  - Bluetooth (IEEE 802.15.1)
  - Zigbee (IEEE 802.15.4)
  - 6LoWPAN (IEEE 802.15.4, IPv6)
- Named TIM or WTIM (Wireless Transducer Interface Module)
- Many options, including TIM to TIM com. via NCAP
Dot 0 TEDS Compiler

- Part of Ph. D. thesis of Wai Liu (Univ. at Buffalo)
# Dot 0 Command/Response Structure

<table>
<thead>
<tr>
<th>Byte Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Destination Transducer Channel Number (Most significant byte)</td>
</tr>
<tr>
<td>2</td>
<td>Destination Transducer Channel Number (Least significant byte)</td>
</tr>
<tr>
<td>3</td>
<td>Command Class</td>
</tr>
<tr>
<td>4</td>
<td>Command Function</td>
</tr>
<tr>
<td>5</td>
<td>Length (Most significant byte)</td>
</tr>
<tr>
<td>6</td>
<td>Length (Least significant byte)</td>
</tr>
<tr>
<td>7-N</td>
<td>Command dependent bytes</td>
</tr>
</tbody>
</table>

## NCAP Command Message Structure

<table>
<thead>
<tr>
<th>Byte Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Success/Fail Flag</td>
</tr>
<tr>
<td>2</td>
<td>Length (Most significant byte)</td>
</tr>
<tr>
<td>3</td>
<td>Length (Least significant byte)</td>
</tr>
<tr>
<td>4-N</td>
<td>Reply dependent bytes</td>
</tr>
</tbody>
</table>

## TIM Reply Message Structure
Wireless Dot4 TEDS

- Large TEDS (Dot 0) takes too much time and energy to transmit
- Small IEEE 1451.4 TEDS is preferable and compatible with SNAP format
- However some information is lacking and must be made up somewhere for full Dot0 compatibility
- We do the conversion in the gateway so gateway is an NCAP.
Dot 4 TEDS -- TEDS only

- UUID (Universal **Unique** Identifier)
  Supplied by EEPROM (DS2433) manufacturer (6 bytes)

- Basic TEDS (8 bytes)
  - Model Number (15 bits)
  - Version Letter (5 bits, A-Z)
  - Version Number (6 bits)
  - Manufacturer ID (14 bits)
  - Serial Number (6 bits)

- IEEE Template or Manufacturer’s TEDS
  Sensor type and calibration parameters (32 bytes)

*Conversion to Dot 0 TEDS possible (but not unique)*
IEEE 1451 as a Universal Digital Sensor Base Format

- Specialized networks can handle only a limited number of sensor types or uses non-compact format
  - 1451 is much superior at the sensor end
- Most applications require individualized displays or graphical user interfaces – 1451 is a fixed format and poorly suited at the user end
- Network oriented applications prefer XML or similar formats which are convenient, but are too verbose at the sensor end
- 1451 at the sensor end (Sensor Fusion level 0) combined with translators is the best solution.
Summary

- Features of low-power wireless sensor were described
- Need for compact format discussed
- Advantages of IEEE 1451 explained
- Wireless temperature sensor example given

Contact: designer@eesensors.com
End

- Backup Slides Follow

www.eesensors.com
Status of Various Parts of IEEE 1451

- **1451.0** – Basic data/TEDS format **Done (2007)**
- **1451.1** – NCAP/Computer Interface **Done (1999)** * Needs revision
- **1451.2** – RS-232 **Done (1997)** * Needs revision
- **1451.3** – Wired Multi-drop **Done (2002)** * Needs revision
- **1451.4** – TEDS Only **Done (2005)**
- **1451.5** – Wireless (WiFi, Zigbee, etc) **Done (2007)**
- **1451.6** – CAN Bus **In process**
- **1451.7** – RFID **In process**

* Most used

* Needs revision
IEEE 1451 – the Universal Transducer Language

- Problem: too many network protocols in common use
- Narrow solutions and borrowed protocols have not worked
- Sensor engineers in the fragmented sensor industry need a simple method of implementation

- How can it be done?
- We need something like USB, except for sensors
- Solution: the IEEE 1451 Smart Transducer Protocol open standard is the best universal solution
- Supported by NIST, IEEE and many Federal agencies
Harmonization Meeting Summary
(Held at NIST Quarterly)

- 25-40 attendees, mostly government
- DOD, DHS, DJ, DS represented (also NASA subcontractor)
- All working under directives to implement standards
- DHS new directive requires new sensors to use existing open standards if available (not proprietary or invent new)
- DOD joint task group working on standards and expects to support test bed. Possible FY’10 requirement.
- Several test beds involving IEEE 1451 started (Esensors has some part in all).
Harmonization of IEEE 1451 with Internet sensor standards
Comments on Wireless Sensor Networks

- Currently fashionable in academic and VC circles
- Recently available low-cost, high performance RF transceiver chips greatly simplify design
- Improvements in embedded microcomputers and smart sensor design also make wireless sensors much more practical now
- Can be lower cost and easier to use than wired sensors
- Point-to-point and mesh protocols available.
- Specialized sensor networks (e.g. Zigbee) much more efficient for wireless sensors, especially battery operated.
- Lack of standards inhibiting growth of industry.
Networked Sensor Block Diagram

SMART SENSOR

Parameter in

Sensor Element → Signal Conditioner → A/D → Micro Computer

ID
Calibration

Tx

Network Interface → Network

From Smart Sensor

To others

PC or Server → Internet

To

SAS -- Wireless with SNAP/1451
Sensor/Transducer Networks

- A network connects more than one addressed sensor (or actuator) to a digital wired or wireless network.
- Both network and sensor digital data protocols are needed.
- Standard data networks can be used but are far from optimum.
- Numerous (>100) incompatible sensor networks are currently in use – each speaking a different language.

The Tower of Babel
Serial Bus Format
and Relation to other Networks

- Tester uses RS232 serial bus only but…
- Interfaces to other physical devices (USB, RS485, Bluetooth, Zigbee, ….) available.
- TEDS retrieval is one feature
- Sensor data read (protocol check) for each channel:
  - **Idle mode** – full scale value of sensor reading
    (Checked against TEDS, error flag is not correct)
  - **Operating mode** – actual sensor reading
    (Must be within sensor range)
TEDS Sections Implemented

- Meta TEDS
- Meta ID TEDS
- Transducer Channel TEDS
- Transducer Channel ID TEDS
- Calibration TEDS
- Calibration ID TEDS
- XdrcName TEDS

Referenced by TEDS section/access code (e.g. #1 for Meta-TEDS)
Meta-TEDS Writer Screen

META TEDS

Change Default Value as Desired

Enter ZIPCODE For UUID

14228

Number of Implemented Transducer Channels

1

Operational Time-Out (Sec)

1.0

Slow Access Time-Out (Sec)

1.0

Self-Test Time (Sec)

1.0

Using Control/Vector/Proxy Groups

NO

NEXT

COPYRIGHT @ 2005 Wei Liu, University at Buffalo. All rights reserved
Channel/Calibration TEDS
(for linear sensors)
TEDS Reader

TEDS Reader

Access Code 3

CHANNEL TEDS

TEDS Analysis Result:

TEDS HEADER: Calibration key (Header Field: 10)
Number of Bytes: 1
DATA: 01

TEDS HEADER: TransducerChannel type key (Header Field: 11)
Number of Bytes: 1
DATA: 01

Lower range limit (Header Field: 12)
DATA: 80 80

Upper range limit (Header Field: 13)
DATA: 80 80
IEEE 451 TIM Compliance Tester

- TIM (Transducer Interface Module) is most complex and done by sensor design engineers
  (TIM tester can be used by the few NCAP designers)
- Tester verifies compliance of a TIM to IEEE 1451.0 (Dot 0) protocol
- Focus is on TEDS checking and data transfer format
- Physical device compliance not checked (part of other standards, e.g. RS485, Bluetooth)
- Tester uses serial bus (RS232)
- Testing may be done by Internet
Network side (NCAP) options (wired)

- Internet/Ethernet
- PC Readout
- Industrial network

All use Dot 0 protocol
Data Readout Examples
(via Internet)

- Sensor data converted to ASCII for display

- TEDS data is displayed in hexadecimal form
Prototype TIM and NCAP

- NCAP interfaces to Internet via Ethernet
### Standard Transducer Units

#### SI Based Units

<table>
<thead>
<tr>
<th>Base Quantity</th>
<th>Name</th>
<th>Unit Symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length</td>
<td>meter</td>
<td>m</td>
</tr>
<tr>
<td>Mass</td>
<td>kilogram</td>
<td>kg</td>
</tr>
<tr>
<td>Time</td>
<td>second</td>
<td>s</td>
</tr>
<tr>
<td>Electric current</td>
<td>ampere</td>
<td>A</td>
</tr>
<tr>
<td>Thermodynamic temperature</td>
<td>Kelvin</td>
<td>K</td>
</tr>
<tr>
<td>Amount of substance</td>
<td>mole</td>
<td>mol</td>
</tr>
<tr>
<td>Luminous intensity</td>
<td>candela</td>
<td>cd</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Data Type</th>
<th>Number of octets</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Physical units interpretation</td>
<td>UInt8</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>(2 * &lt;exponent of radians&gt;) + 128</td>
<td>UInt8</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>(2 * &lt;exponent of steradians&gt;) + 128</td>
<td>UInt8</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>(2 * &lt;exponent of meters&gt;) + 128</td>
<td>UInt8</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>(2 * &lt;exponent of kilograms&gt;) + 128</td>
<td>UInt8</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>(2 * &lt;exponent of seconds&gt;) + 128</td>
<td>UInt8</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>(2 * &lt;exponent of amperes&gt;) + 128</td>
<td>UInt8</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>(2 * &lt;exponent of kelvins&gt;) + 128</td>
<td>UInt8</td>
<td>1</td>
</tr>
<tr>
<td>9</td>
<td>(2 * &lt;exponent of moles&gt;) + 128</td>
<td>UInt8</td>
<td>1</td>
</tr>
<tr>
<td>10</td>
<td>(2 * &lt;exponent of candelas&gt;) + 128</td>
<td>UInt8</td>
<td>1</td>
</tr>
</tbody>
</table>
Future of Networked Sensors

- Computer-based monitoring and control applications are increasing in commercial, industrial and military sectors.
- Networked, and often wireless, sensors offer performance and cost advantages over traditional methods.
- Wider use of networked sensors is inhibited by lack of standards (especially National scale networks, wireless and multi-vendor, long-term installations).
- I expect sensor standard and harmonization efforts to succeed within 2 to 5 years and Federal Agencies (DHS, DOD; NASA) to favor IEEE 1451.
Future Prospects for IEEE 1451

- There has been little interest in previous parts of IEEE 1451 (Dots 2, 1 & 3), except for Dot 4 in certain areas.
- The basic Dot 0 (and Dot 5) are more carefully composed and thus likely to be accepted.
- The compiler may answer the complexity issue but still implementation of any full-featured standard will be difficult.
- The US government may mandate a sensor data standard and the NIST-supported IEEE 1451 is the most recognized candidate.
- The sensor industry, especially the wireless network sector, must recognize the business advantages of a single sensor data standard.
Dot 4 TEDS Writer and Reader (PC Screens)
TIM Tester – Data retrieval

IEEE 1451 TIM Tester V.1.0

Query Channel TEDS Analysis:
Result: Supported

Sensor Type: Temperature Sensor
SI Unit: K
Low Limit Value: 233
High Limit Value: 353

Query Calibration TEDS Analysis:
Result: Supported

Slope Constant Value: 1
Intercept Constant Value: 273.15
Sensor Data Value: 26.36719
Calibrated Sensor Value (in SI Unit): 299.5172

Operating Mode:
COM 1

Start Test
RFID with Sensors (Dot 7)

- Combines Radio Frequency IDentification with sensors
- Uses standard (ISO/IEC) RFID communication/format with additional sensor memory section
- Sensor format is based in IEEE 1451 protocol
- Typical application is tracking and monitoring perishable shipments (temperature, shock/vibration)