



DA SYSTEMS

Update

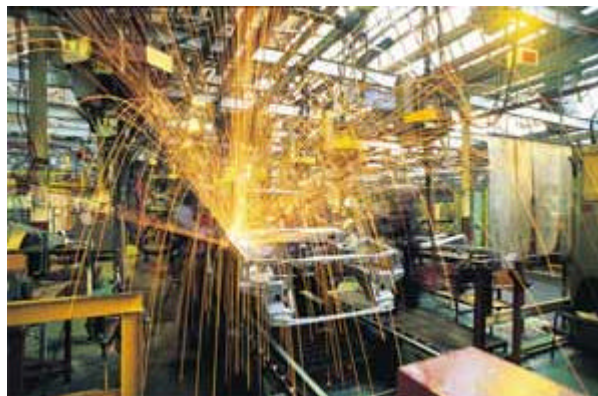
IEEE-1451.2

Here's a status report on the transducer-to-microprocessor communications protocols and the transducer electronic data sheet formats that make up the IEEE effort to network smart sensors.

Robert N. Johnson,
elemonitor, Inc.

We live in an age where connectivity and information flow are becoming increasingly important. Smart sensors and smart actuators (collectively called smart transducers) can help by converting data into information and reducing communications bandwidth requirements. These devices can also provide self-identification, self-testing and adaptive calibration, ease of setup, and improved rejection of spurious inputs.

Given these advantages, why aren't smart transducers in wider use? One reason is that there are just too many network architectures and protocols for transducer manufacturers to support. The IEEE-



1451.2 standard was the first of a planned family of standards for connecting smart transducers to networks. These standards will make it possible to develop network-capable and network-independent plug-and-play transducers for use in embedded products, networked controllers, and distributed data acquisition (DA) systems.

The IEEE-1451 Family

The proposed IEEE-1451 family of standards covers network-capable smart transducers from the interface to the transducer to a high-level

object model of behavior, attributes, and data communications.

The basic functional boundaries and interfaces defined in the proposed standards are illustrated in Figure 1. The architecture consists of a

network, a network-capable application processor (NCAP), and a smart transducer interface module (STIM). Note that the transducers themselves are considered part of the STIM. In fact, to provide the critical self-identification features, the transducer must be inseparable from the STIM electronics during normal use. For the basic system definition, the intended field break is the transducer-independent interface (TII) between the STIM and NCAP.

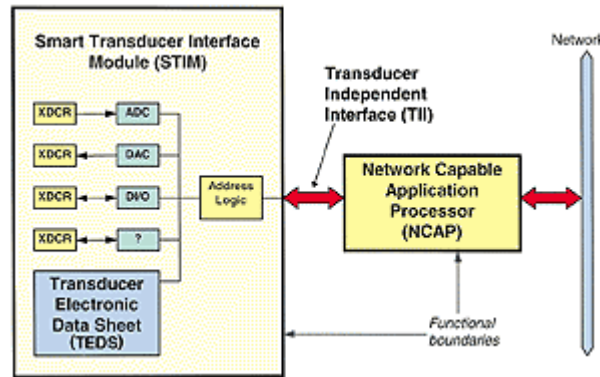


Figure 1. IEEE-1451.2 defines not only the electrical interface between a smart transducer and a network node but also a common functional model for the transducer as well as an embedded machine-readable data sheet.

The high-level object model consists of a series of functional blocks that can be plugged into a logical backplane to form a working system (see Figure 2). The blocks attached to the backplane make up the NCAP portion of the smart transducer. The transducer interface shown in Figure 2 refers to the TII. The physical block represents the physical elements of the smart transducer and is always present. The network block is optional, but any network can be supported in an NCAP by replacing the network block. Functional blocks can be included to provide special-purpose functions, such as Fourier transforms or proportional integral differential (PID) control loops. One or more transducer blocks are used to present an abstract view of the transducers to the processor and to other nodes on the network. Each transducer block supports a TII to which a STIM can be connected.

The IEEE-1451.1 standard, with the subtitle known as "Network Capable Application Processor (NCAP) Information

Model," has been approved by a ballot of IEEE members and is being prepared for publication by IEEE headquarters. The IEEE-1451.2

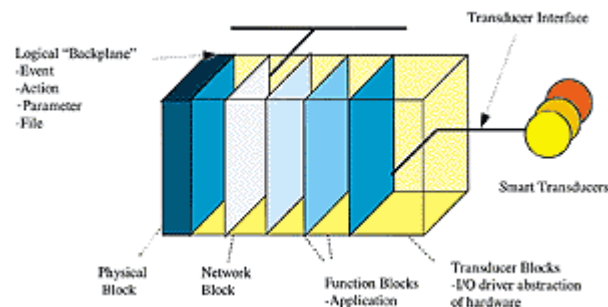


Figure 2. The recently approved IEEE-1451.1 standard expands the common functional model of a transducer and defines an object-oriented view of the device's methods and attributes. The abstract model is based on a virtual backplane into which are plugged hardware and software blocks to provide the desired functionality. The transducer interface is expected, but not required, to be the one defined in IEEE-1451.2.

standard, with the subtitle known as "Transducer to Microprocessor Communication Protocols and Transducer Electronic Data Sheet (TEDS) Formats," was approved September 1997. This covers the STIM, the TII, and the portion of the NCAP transducer block that communicates directly with the STIM.

IEEE-1451.2 defines other critical elements of smart transducer operation and of the communications interface. These include the formats for the TEDS, the transducer function type (e.g., sensor, actuator, buffered sensor, and event sensor), and a general-purpose calibration and correction engine. The normal data format for an IEEE-1451.2 smart transducer is an IEEE-754 floating-point number using standard SI units, although provision has been made for using raw A/D or D/A converter counts for systems that must wring the last drop of performance out of the available hardware.

The Electrical Interface

The physical connection between the NCAP and the STIM is the 10-pin TII. The TII is built around synchronous serial communications based on the serial peripheral interface (SPI) protocol. Several dedicated lines have been added to provide power, ground, and special-purpose control lines. The recommended TII pin assignments are listed in Table 1, which also indicates whether each physical signal is considered an input or an output by the NCAP and the STIM.

Table 1

Recommended TII Pin Assignments				
Pin number	Signal Name	Wire Color	Direction for NCAP	Direction for STIM
1	DCLK	brown	OUT	IN
2	DIN	red	OUT	IN
3	DOUT	orange	IN	OUT
4	NACK	yellow	IN	OUT
5	COMMON (GROUND)	green	POWER	POWER
6	NIOE	blue	OUT	IN
7	NINT	violet	IN	OUT
8	NTRIG	gray	OUT	IN
9	POWER (5 VDC)	white	POWER	POWER
10	NSDET	black	IN	OUT

The TII logical signal definitions and functions are shown in Table 2, which also indicates whether a signal is positive or negative logic and whether it is level or edge sensitive.

Note that the input and output naming conventions for the logical signals are from the point of view of the STIM.

NINT is the only TII signal line that the STIM is permitted to assert at will. Other than that, the NCAP controls all communications, and all message exchanges are originated by the NCAP. The other STIM outputs are not really under the full control of the STIM. NSDET is a passive signal (pin is tied to common) used to detect the presence of a STIM, and NACK is only asserted in response to an action by the NCAP. The NCAP is the SPI communications master and always controls DCLK. Even in the case of NINT, which might have been more accurately named service request instead of interrupt, the resulting communications are under the control of the NCAP, which is under no timing constraint as to when it responds.

Table 2

TII Signal Definitions			
Line	Logic	Driven by	Function
DIN	positive logic	NCAP	Transports address and data from NCAP to STIM
DOUT	positive logic	STIM	Transports data from STIM to NCAP
DCLK	positive edge	NCAP	Positive-going edge latches data on both DIN and DOUT
NIOE	active low	NCAP	Signals that the data transport is active and
z	z	z	delimits data transport framing
NTRIG	negative edge	NCAP	Performs triggering function
NACK	negative edge	STIM	Serves two functions:
z	z	z	1. trigger acknowledge
z	z	z	2. data transport acknowledge
NINT	negative edge	STIM	Used by the STIM to request service from the NCAP
NSDET	active low	STIM	Used by the NCAP to detect the presence of a STIM
POWER	N/A	NCAP	Nominal 5 V power supply
COMMON	N/A	NCAP	Signal common or ground

For a simple transducer, the NCAP uses NTRIG to control when new data are taken (in the case of a sensor) or are acted on (in the case of an actuator). In other words, a simple sensor takes data only when triggered by the NCAP, and a simple actuator updates its output only when

triggered by the NCAP. More complex transducer function models (e.g., buffered sensors, data sequence sensors, and event sensors) make more complex use of NTRIG.

Communications Protocol

From the point of view of the NCAP, the STIM looks like a memory device. All data and commands are accessed by way of a functional address, which is simply the combination of the requested service plus the channel to which it is addressed. Global communications to the STIM as a whole are addressed to channel 0. This is why a STIM is limited to 255 transducer channels instead of 256.

The basic protocol for a communication between the NCAP and the STIM is that the NCAP clocks a functional address into the STIM using the DIN and DCLK signal lines. For a write, the NCAP keeps clocking DCLK and places the data on DIN. For a read, the NCAP keeps clocking DCLK and looks for the data on DOUT. For all communications, NIOE serves as a chip select to tell the STIM that the data transport function is active. The STIM uses the NACK to acknowledge data bytes and the trigger signal. To keep the interface protocol simple, data communications and trigger functions cannot be used at the same time.

As originally written, the draft standard did not require a STIM to acknowledge each byte. This meant that the DCLK rate had to be slow enough to allow the STIM to prepare to receive or transmit the next byte within a single DCLK period. Later, the IEEE changed this to require NACK to acknowledge each byte. This was supported by early experience with prototype NCAPs and STIMs, requiring acknowledgment of each byte by the NACK. This change had a dramatic effect on the achievable data transfer rate, allowing a 100:1 increase in rate for some prototype STIMs.

Power Connections

In addition to the logical signals described above, the TII also supplies power and a common ground reference to the STIM. All the TII lines--including power and common--must be isolated from frame or earth ground by the STIM.

The NCAP supplies up to 75 mA at 5 V. The standard provides for supplemental power, independent of the NCAP, if needed for sensitive or high-power transducers. But only the NCAP can provide power for the STIM interface control circuitry. These power and ground isolation requirements are included to reduce the noise transmitted to the NCAP by the STIM and to minimize the potential for ground loops.

Physical Connector

The description of the 10-pin TII seems to be missing one element--the physical connector. The IEEE-1451.2 working group attempted to standardize on a connector, but it discovered that connectors were just too application dependent to allow a consensus. The applications of interest range from very cost sensitive to very high performance. This is

one area in which the near-universal applicability of the IEEE-1451.2 standard was almost a disadvantage.

The working group anticipates that the industry will develop de facto standards for primary applications. Companies have successfully used simple 5 by 2 headers and ribbon for demonstration systems and for benign environments, such as applications where the TII is contained in a protective enclosure. Other connectors--such as a 15-pin subminiature D-shell, which is commonly used for computer video cables--are perhaps more appropriate for industrial environments.

TEDS Formats

The standard defines eight TEDS formats. Of these, two are required, and the others are optional. The two required structures and two of the optional ones are defined as binary data formats and are machine readable only. The other TEDS structures are human readable and are stored as strings in one of several different character sets. The standard enumerates 18 different languages using 21 international character sets and makes provision for the addition of more as they are identified.

The working group spent much of its time defining the TEDS structures and placed considerable emphasis on identifying all useful data fields for inclusion in the standard. It also made provision for expanding the formats by using industry extensions. Each TEDS structure includes a length (i.e., the byte count) and a check sum for error detection. A brief description of the TEDS formats follows.

Meta-TEDS Data Block (Required and Machine Readable). This block contains data that describe the STIM as a whole, such as revision levels, extensions, worst-case timing values, and number of channels. One interesting feature of the Meta-TEDS is the unique STIM identifier. The working group wanted to define a system for identifying each STIM from any manufacturer without having to resort to a coordinating committee to assign codes. The resulting identifier is a combination of the latitude and longitude of the manufacturing facility and the UTC code for the time of manufacture. A manufacturer can assign codes to different product lines by using slightly different coordinates for each line. The only requirement is that the coordinates must represent locations that are under the physical control of the manufacturer.

Channel TEDS Data Block (Required and Machine Readable). The Channel TEDS defines the functional model, calibration model, physical (SI) units, upper and lower limits, timing restrictions, and any other data needed to describe the functioning of each transducer channel. If more than one channel is implemented, the structure is repeated for each one.

Calibration TEDS Data Block (Optional and Machine Readable). If you're going to use the correction engine, you have to include a Calibration TEDS to define the coefficients to be used. In addition, this data block defines the last calibration date and time, as well as the required calibration interval for each channel.

Meta-Identification TEDS Data Block (Optional and Human Readable).

The Meta-Identification TEDS provides a human-readable version of the identification data for the overall STIM. The information is repeated once for each language used. String fields include the manufacturer's name, the model number, the serial number, version codes, date codes, and a product description.

Channel Identification TEDS Data Block (Optional and Human Readable). This TEDS includes information similar to the Meta-Identification TEDS, except that it's for an individual channel. This is useful when a STIM built by one company contains transducers manufactured by one or more different companies.

Calibration Identification TEDS Data Block (Optional and Human Readable). The Calibration Identification TEDS provides a human-readable description of information deemed relevant to the calibration of each channel. The structure is repeated for each channel and for each language supported.

End Users' Application-Specific TEDS Data Block (Optional and Human Readable). This data block is additional storage for human-readable data not covered by the specific TEDS fields described above. An example of an end-user field might be a description of the location of the STIM or even the name and telephone number of the person to call for service. As for all of the human-readable TEDS, the fields are repeated for each language supported.

Generic Extension TEDS Data Block (Optional, Readability Not Specified). This data block lets you add extensions to a TEDS. Each extension will be defined in a manner similar to that used in the IEEE-1451.2 standard, and each originating organization will have to apply to the IEEE Standards Office for a TEDS extension ID number. A special number is reserved for programming and testing prototypes, which include experimental TEDS extensions.

Correction Engine

The last feature of the IEEE-1451.2 standard to be examined here is the correction engine. Correction is the application of a mathematical function on transducer data from one or more STIM channels and/or data delivered from other sources. In other words, the correction engine takes the output of one or more transducer channels plus any other data that may be necessary and uses a mathematical formula and one or more stored coefficients to produce a corrected value for the channel. The output of the correction engine is usually a floating-point number in SI units ready for communication to a user or client process. And bear in mind that the correction engine is reversible, in that it can also be used to convert a floating point input command for an actuator to an integer representing the required raw control input to the actuator.

The correction engine defined for IEEE-1451.2 uses a multinomial (multivariate polynomial) function with a virtually unlimited degree of the polynomial. To limit the degrees of the input to the correction engine, the polynomials can be segmented: different sets of coefficients can be used between specified limits of the input values.

The correction engine is powerful and can provide a standard way of describing the calibration constants and correction coefficients for a broad range of sensors and actuators. However, use of the correction engine is optional. A system with limited computational capability can use a simple integer data representation rather than having to deal with floating-point mathematics.

Conclusions

The IEEE-1451.2 standard is here at last, allowing transducer manufacturers to produce network-capable and network-independent smart transducers without having to become experts in each network and without having to decide how many of each network-dependent version of their products to manufacture and stock. With IEEE-1451.2, all transducers are identical, regardless of the target control network or fieldbus.

An additional but sometimes overlooked benefit is that the standard will let system integrators upgrade control networks without having to change the sensors. With IEEE-1451.2, the transducer manufacturers and the system integrators will have more freedom from the details of the fieldbus implementations.

The standard has been implemented by more than one company, and intercompany hardware and communications compatibility has been demonstrated. Now you can buy commercial hardware that is tailored for specific real-world applications.

Robert N. Johnson is President of Telemonitor, Inc., 9055F Guilford Rd., Columbia, MD 21046; 410-312-6621, fax 410-312-6692, robertj@telemonitor.com

[We Love Feedback](#)

Sensors® and Sensors Expo® are registered trademarks of [Advanstar Communications Inc.](#)
[Privacy Policy](#)

[Sensors Online Home](#) | [Sensors Expo](#) | [Contact Us](#)

SENSORS
Online