XVII IMEKO World Congress
Metroloy in the 3rd Millennium
June 22–27, 2003, Dubrovnik, Croatia

AN APPLICATION OF THE IEEE 1451.2 CORRECTION ENGINE IN AN INTEGRATED SENSING STRUCTURE

Helena Ramos[1], O. Postolache[1,2], M. Pereira[2], P. Girão[1]

1. Instituto de Telecomunicações, DEEC, IST, Av. Rovisco Pais, 1049-001, Lisboa, Portugal
2. Escola Superior de Tecnologia, Instituto Politécnico de Setúbal, 2914-508, Setúbal, Portugal

Abstract − This paper reports the implementation of a sensing and alarm integrated system connected through the Transducer Independent Interface (TII) to a Network Capable Application Processor (NCAP), emulated in a PC. The “Correction Engine” and the calibration features of the NCAP are used to process the data acquired via the transducer and also to add auto range capabilities to the conditioning circuits of the input quantities. The NCAP is also used to establish the interface with the Internet. A working prototype is developed for water quality monitoring with pH and temperature sensors and with an alarm to detect values outside a pre-defined range.

Keywords: smart sensor, IEEE 1451, calibration.

1. INTRODUCTION

The recently approved IEEE 1451 standard aims at simplifying transducer connectivity to existing networks. Fig. 1 illustrates a typical device based on the two subprojects of the standard already approved. One of the documents, 1451.2[1], defines the Smart Transducer Interface Module (STIM), which includes the transducers and the signal conditioning that may be needed, a non-volatile memory called Transducer Electronic Data Sheet (TEDS) and the Transducer Independent Interface (TII) that enables communication between the STIM and the Network Capable Application Processors (NCAP). This module, defined in standard IEEE1451.1[2], implements the network architecture, the correction of raw data coming from the STIM and may include other data processing and control functionality applications.

On the other end, nowadays microprocessor-based sensing systems are replacing traditional analogue measurement systems. The programmable interface controller [3] is an attractive solution due to its low cost, integrated analogue to digital conversion capabilities and it allows an easy way to interface with higher-level systems.

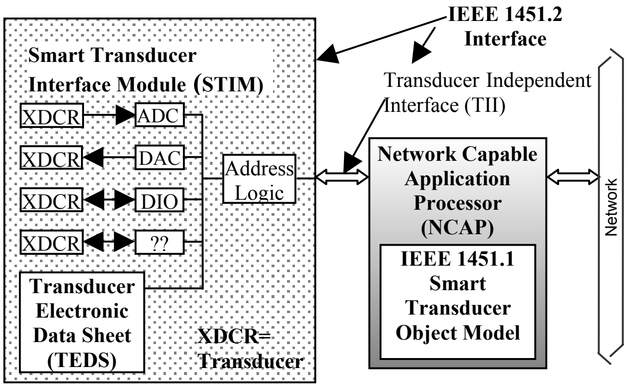


Fig. 1. The IEEE1451.2 version of a smart sensor architecture

The purpose of this work is the use of a microcontroller (PIC 16F877) to implement an integrated sensing and alarm system taking advantage of some of the IEEE1451 standard features. The micro is used to interface a NCAP simulated in the PC and its analogue to digital capabilities are an important tool to convert input data. A data memory EEPROM keeps a register of the transducers information. The correction engine of the NCAP performs correction of raw data for non-linear effects of a pH probe and the effect of temperature on sensor’s behaviour. Experimental data show clearly that a correction based only on Nernst law is not sufficient due to pH variation of the buffer solutions and to other nonlinear effects. The NCAP that implements Internet access is also used to set the gain of the differential amplifiers used in the conditioning circuit of the pH probe and the alarm.

2. SYSTEM DESCRIPTION

The architecture of the system consists of a measurement and communication component, a Transducer Independent Interface (TII) and a virtual Network Capable Application Processor (NCAP) in a PC.

2.1. Measurement and Communication Station

The measurement and communication station (MCS), represented in Fig. 2 contains two sensors and an alarm, the signal conditioning circuits, and the microcontroller 16F877 with an analogue to digital (A/D) converter included.

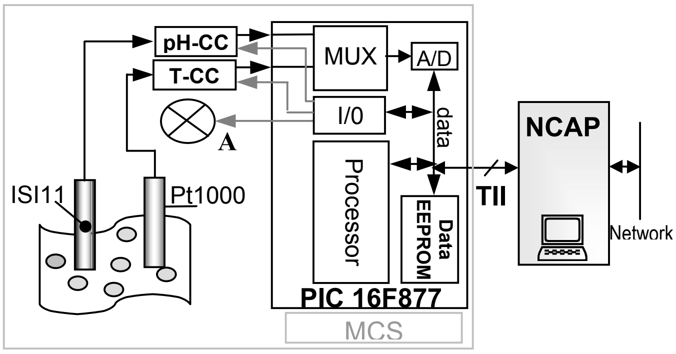


Fig. 2. Measurement system block diagram: MUX-multiplexer; A/D-analog-to-digital converter; I/O-digital input/output; pH-CC-pH conditioning circuit; T-CC-temperature conditioning circuit; A-alarm.

The pH of the water is measured using a ISI11 pH transducer that delivers a voltage between 0-300mV for pH values in the 0 to 14 range. The extremely high output impedance of the pH probe (150MΩ) implies the use of a precision operational amplifier such as MAX406, with a supply current of less than 1,2μA as a buffer circuit. An instrumentation amplifier (AD524) with programmable gain is then used to adjust the voltage to the input range of the A/D converter. The gain is set automatically using the correction model. The temperature sensor is a resistive temperature detector Pt1000. The conditioning circuit is a 0.14mA constant current source (based on LM324) implemented in order to obtain a voltage V_T linearly dependent on temperature.

Alarm events are signalled if pH drops or exceeds given alarm set-points or if the derivatives of pH and temperature exceed threshold values.

2.2. *Network Capable Application Processor: NCAP*

The virtual NCAP includes as hardware component a PC with Ethernet connection and as software component was fully developed in LabVIEW 6.1. Network communication capabilities already exist in LabVIEW, thus only the STIM block has to be implemented. The implemented block is composed by: the STIM driver, responsible from getting data across the interface; the TEDS parser that has the knowledge about the 1451.2 TEDS structure and reconfigures data; the 1451.2 Application Programming Interface (API), which provides access to TEDS blocks, sensor readings, actuator control, interrupt requests and triggers; a "correction engine" to operate on the registers containing the raw transducer values.

The hierarchy structure of a graphical programming language like LabVIEW makes easy to build Virtual Instruments (VI) that perform standardized functions as "ReadSensor" to read data from the sensor or "WriteControlCommand" to change sensor parameters. The set of all the functions of the STIM driver is in a library, represented in Fig. 3 and each of them is associated to a specific function.

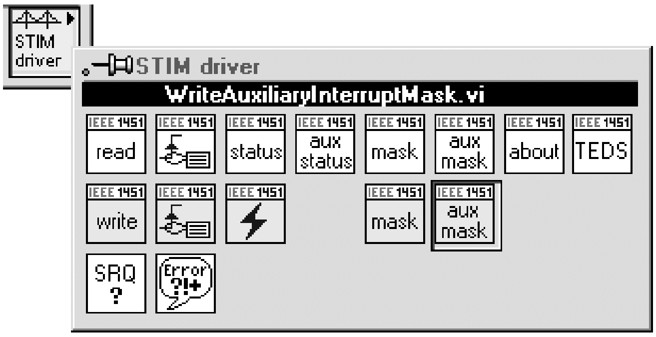


Fig. 3. STIM driver library implemented in LabVIEW

This library is made up of medium level functions that establish the connection between the LabVIEW programmer and the dynamic link library (DLL) used to control the Transducer Independent Interface (TII). Fig. 4 shows this implementation in two layers that enables the programmer to build applications in LabVIEW, shielding the access to the interface.

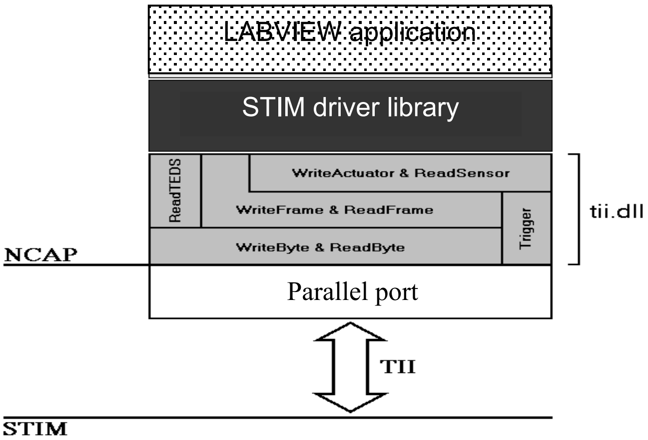
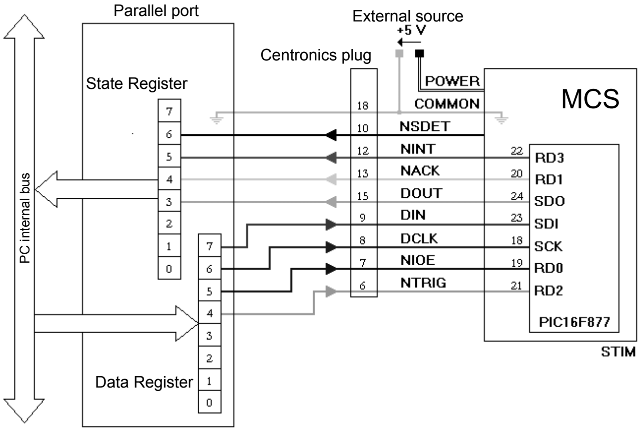


Fig. 4. STIM library context

2.3. *Transducer Independent Interface: TII*

To communicate with the NCAP the 10-pin Transducer Independent Interface (TII), defined in the 1451.2 standard, is built using a synchronous serial data transfer based on the Serial Peripheral Interface (SPI) protocol integrated in the PIC microcontroller (16F877) used in the Measurement and Communication Station (MCS) and the Centronics port of the PC. In Fig. 5 the connections between the PIC16F877 and the PC internal bus are depicted. The Data Register of the parallel port of the PC is used as an output as it specifies the state of the lines that excite the microcontroller and the State Register is used to read the signals.



Fig, 5. Connections between the MCS and the NCAP

As stated before, a DLL developed in C++ implements the set of functions used to control the parallel port of the PC in order to build the Transducer Independent Interface (TII). The protocols, timing and electrical specifications are in accordance with the Standard 1451.2.

2.4. *Transducer Electronic Data Sheet: TEDS*

The Transducer Electronic Data Sheet (TEDS) is one of the main technical innovations introduced with the standard. It is a non-volatile memory used to store parameters describing the STIM itself and any transducer associated with it.

In the implementation reported in this paper the TEDS is stored in the program address space of the PIC microcontroller. It contains the two TEDS machine-readable strutures that are mandatory: the Meta-TEDS and the Channel TEDS and an optional Calibration TEDS to be used by the correction engine. As defined in the IEEE 1451.2, the Meta-TEDS data block describes the STIM as a whole, including revision levels, an unique identifier, worst-case timing values, the number of channels and channel grouping and the Channel TEDS defines the functional model, calibration model, physical units, upper and lower limits, timing restrictions and any other data needed to describe the functioning of each transducer channel. In the STIM represented by the Measurement and Communication Station-MCS described in *2.1.*, there are three Channel TEDS strutures concerning respectively the pH sensor, the temperatre sensor and the alarm and one Calibration TEDS that stores the coefficients to be used in the correction engine.

2.5. Communication

The NCAP controls all communications and all message exchanges, except for the request service (NINT line in Fig.4). Data and Commands are accessed using a functional address that includes the requested service and the address of the addressed channel. As a master device, the NCAP asserts data out on the falling edge of the clock (DCLK line), and latches data on the rising edge of the clock.

2.6. Correction Engine

The correction engine is an algorithm that converts raw readings coming from the transducer-side, which frequently are expressed into A/D or D/A converter counts, to the desired user units, corrected for calibration errors or even temperature compensated.

The correction engine is not mandatory within the standard IEEE 1451 and it isn't even specified where the conversion should take place. Running the correction process in the STIM means that the transducers will cost more and consume more power. However, it can be a useful solution for large arrays of transducers or for very high-speed transducers. If the conversion is to be performed in the NCAP, the channel correction coefficients are uploaded from the calibration TEDS and made available to the NCAP and the conversion into the user units is performed here. The only disadvantage can be the additional cost of the NCAP processor to support the additional processing. Another solution is to use a host computer connected to the network to compute the conversion. The disadvantage arises in systems with many transducers or in distributed systems.

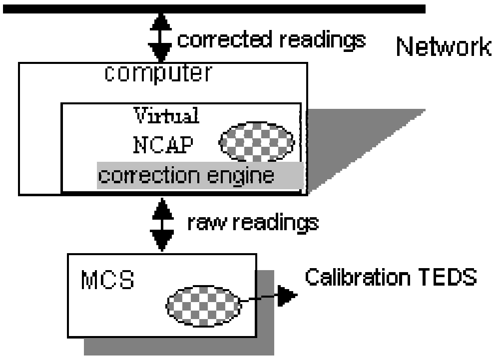


Fig. 6. Correction engine in the NCAP

For the application reported in this paper, as the pH electrode and measurements are temperature sensitive, a separate temperature measurement is required. Fig. 6 shows compensation being performed at the virtual NCAP with the "correction engine" in order to accurately compute the pH value of the sample at a specific temperature.

2.7. The Correction Function

The standard 1451.2 defines the correction function as a multivariate polynomial of the form:

$$\sum_{i=0}^{D(1)}\sum_{j=0}^{D(2)}\cdots\sum_{p=0}^{D(n)} C_{i,j,\dots,p}\,[X_1-H_1]^i\,[X_2-H_2]^j\cdots[X_n-H_n]^p \qquad (1)$$

where X_n represent the input variables, H_n the offsets of the input variables, D(k) the degree of the input X_k and $C_{i,j,\ldots,p}$ the correction coefficients for each term. As stated before D(k), $C_{i,j,\ldots,p}$ and H_n are data obtained from the calibration TEDS [4].

For the implemented Measurement and Communication Station, there is one actuator channel and two sensor channels. The actuator channel asserts the alarm and has a Calibration TEDS associated with it. The temperature channel provides temperature values (U_T) and the pH channel provides the readings from the pH probe (a voltage, U_{pH}, in response to a pH value) and has also a Calibration TEDS.

For the alarm, a segmented zeroth order conversion is performed. (1) for this conversion is:

$$Y = C \qquad (2)$$

The output, *C*, takes one of the two states (0 or 1) depending on the level of the input (pH value). Table 1 represents the TEDS for the alarm of the MCS.

Number of correction input channels	1
Correction input channel list (actuator channel)	1
Correction input channel–key list	1
Channel degree	0
Number of segments	2
Segment boundary values. (three entries: two lower limits and an upper limit for the last segment)	4 8 14
Segment offset values (two segments imply two offsets)	4 8
Coefficients (zeroth order implies one coefficient for each segment)	0 1

Table 1. Calibration TEDS for the alarm of the MCS

A multiple input conversion is performed for the pH probe: the response of the pH sensor is non-linear and temperature dependent so it is necessary to calibrate it at several temperatures in order to establish a consistent data-reduction methodology. By assigning the voltage probe U_{pH} to channel 2 and a temperature sensor to the U_T channel, the corrected value of pH can be computed. There cases are presented: (a) assumes that the response is is linear with the second variable U_T, and also first order with respect with the primary variable U_{pH}.; (b) in the second case, it is considered a two segment correction for the primary variable. The equation for conversion in both cases is:

$$pH = C_{00} + C_{01} \cdot U_{pH} + C_{10} \cdot U_T + C_{11} \cdot U_{pH} \cdot U_T \qquad (2)$$

(c) The third case, assumes that pH is linear with the second variable and second order with respect to the primary variable. The equation for this conversion is:

$$pH = C_{00} + C_{01} U_{pH} + C_{02} U_{pH}^2 + C_{10} U_T + C_{11} U_{pH} U_T + C_{12} U_{pH}^2 U_T \qquad (3)$$

The correction coefficients are calculated in the calibration procedure, using pH buffer solutions at specific temperatures.

The “correction model” is also used to select the appropriate input amplification or attenuation to keep the signals in an appropriate range for the electronics. This auto-range feature of the pH transducer uses a virtual sensor with zeroth order conversion ($Y=C_0$).

3. RESULTS

The electrical potential difference between the pH measurement and the reference electrodes in the pH sensor depend in a logarithmic manner on the hydrogen ion activity of the solution, a function known as Nernst equation:

$$E = E' + \frac{RT}{nF} \ln \left(a_{H^+} \right) \qquad (4)$$

where $R = 8.31\text{J/(mol.K)}$ is the gas constant, T is the temperature in Kelvin, n the valence of the ion, $F = 96487\text{C}$ is Faraday constant and a_{H+} is the ion activity. In terms of the pH (-log(a_{H+})), equation (1) can be written as:

$$\Delta E = -0.198 \cdot 10^{-3} \cdot T \cdot pH \qquad (5)$$

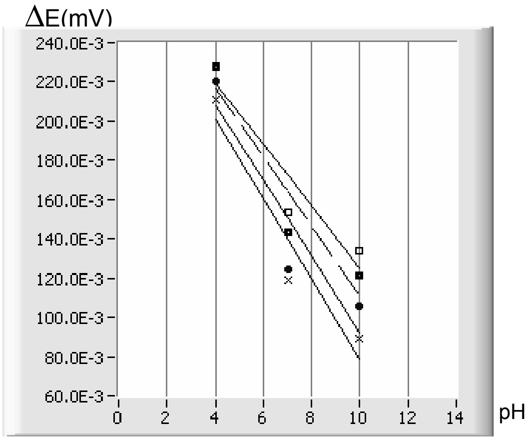


Fig.7. pH probe calibration

Fig. 7 shows the e.m.f. measured with the pH probe in three different buffers with 4, 7 and 10 pH for different temperature values. The significant deviations between the experimental values and the theoretical data based on Nernst law are evident and justify the use of the correction engine to attain a better accuracy in the measured values.

The coefficients of the calibration polynomialsto be placed in the calibration TEDS of the pH channel for cases (a), (b) and (c)are presented in table 2. The calculus of the calibration constants assumes a linear dependence with temperature.

BRIEF DESCRIPTION	(a)	(b)	(c)
Number of correction input channels	2	2	2
Correction input channel list	2	2	2
	3	3	3
Correction input channel–key list	0	0	0
	0	0	0
Channel degree	1	1	2
	1	1	1
Number of segments	1	2	1
	1	1	1
Segment boundary values. (three entries: two lower limits and an upper limit for the last segment)	4	4	4
		7	
		10	
		7	
	10	10	10
	10	10	10
	30	30	30
Segment offset values (two segments imply two offsets)	4	4	4
		7	
Coefficients	C_{00}=18.87 C_{01}=-64.1 C_{10}=-.22 C_{11}=.83	C_{00}=14.67 C_{01}=-45.89 C_{10}=-.17 C_{11}=.62	C_{00}=63.37 C_{01}=-543.2 C_{02}=1231. C_{10}=-1.27 C_{11}=9.49 C_{12}=-16.22
		C_{00}=33.09 C_{01}=-150.7 C_{10}=-.43 C_{11}=.86	

Table 2. Calibration TEDS for the pH channel

The errors calculated as the difference between the e.m.f. measured with the pH probe and the value obtained with (2) and (3) polynomials, for the temperature of 15^{0}C, for the three implemented methods ((a), (b) and (c)) are depicted in Fig. 8.

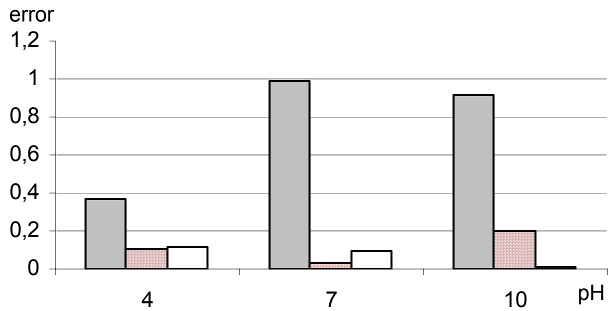


Fig.8. Error obtained for T=15^{0}C

4. CONCLUSIONS

This paper will show the power and versatility of the IEEE-1451.2 correction algorithm.

The results obtained by performing an automated temperature correction show that pH measurements are obtained with errors less than 0.1pH units. Better accuracy could be obtained with the useful range of the probe signal more segmented both in voltage and temperature dependence. As the standard defines a multinomial correction of arbitrary degree, this can be easily achieved just by downloading a new set of coefficients into the TEDS. The number of coefficients in the TEDS settles a commitment between accuracy and speed.

ACKNOWLEDGMENT

This work was supported both by Portuguese Science and Technology Foundation PRAXIS XXI program FCT/BPD/2203/99 and by Project FCT PNAT/1999/EEI/15052. These supports are gratefully acknowledged.

REFERENCES

[1] IEEE Std. 1451.2 IEEE Standard for a Smart Transducer Interface for Sensors and Actuators – Transducer to Microprocessor Communication Protocols and Transducer Electronic Data Sheet (TEDS) Formats, 2000.
[2] IEEE Std. 1451.1 IEEE Standard for a Smart Transducer Interface for Sensors and Actuators – Network Capable Application Processor (NCAP) Information, 2000.
[3] John B. Peatman, "Design with PIC Microcontrollers", *Prentice Hall*, 1998.
[4] Lee H. Eccles, "IEEE-1451.2 Engineering Units Conversion Algorithm", http://www.sensorsmag.com/articles/0599/0599_p107

Prof. Helena Geirinhas Ramos, Instituto de Telecomunicações, DEEC, IST, Av. Rovisco Pais, 1049-001, Lisboa, Portugal.
Phone: +351-218417289; Fax: +351-218417672;
hgramos@alfa.ist.utl.pt
Prof. Octavian Postolache, Escola Superior de Tecnologia, Instituto Politécnico de Setúbal, 2914-508, Setúbal, Portugal
Phone: +351-218417974; Fax: +351-218417672;
octav@popsrv.ist.utl.pt
Prof. Miguel Pereira, Escola Superior de Tecnologia, Instituto Politécnico de Setúbal, 2914-508, Setúbal, Portugal
Phone: +351-218417974; Fax: +351-218417672;
joseper@est.ips.pt
Prof. Pedro Girão, Instituto de Telecomunicações, DEEC, IST, Av. Rovisco Pais, 1049-001, Lisboa, Portugal
Phone: +351-218417974; Fax: +351-218417672;
psgirao@alfa.ist.utl.pt