Normalizing Transducer Signals: An Overview of a Proposed Standard

G. Monte, *Member*, IEEE,
Facultad Regional Del Neuquén
Universidad Tecnológica Nacional
Plaza Huincul, Argentina

Z. Liu, *Senior Member*, IEEE
Toyota Technological Institute
Japan

F. Abate, *Member*, IEEE, V. Paciello, *Senior Member*, IEEE, A. Pietrosanto, *Senior Member*, IEEE
Dep. of Industrial Engineering
University of Salerno
Salerno, Italy

V. Huang, *Senior Member*, IEEE
Better World- IES Standards
USA

Abstract — **This paper presents an overview of the proposed standard ISO/IEC/IEEE P21451-001 for sensor signals that is currently under development. The IoT, Internet of Things, is a reality. Most of things have sensors embedded without drawing distinction between industrial and consumer products. Things can be easily identified and the protocols that govern the traffic over the nets are well defined. However, what is difficult to define is the information that a thing with sensors should report. Therefore, the objective of this standard is to define a first approach to describe sensor signals, in a way that can be applied to any sensor extracting information from samples.**

The paper is divided into two main sections. The first one describes the basic algorithms that extract knowledge from samples. The second one shows the internal structure of the working group for normalizing these algorithms and the remaining challenges.

Keywords — Standards; Sensor Signals; Internet of Things; IEEE 1451; .

I. INTRODUCTION

The world is inferred from sensor signals. Most of things have sensors embedded and communication between those is becoming even more essential. The number of connected devices is growing up faster and, nowadays, there are more things connected to the internet than people on earth [1]. The Internet of Things (IoT), is a new way for technologies to keep in contact with environment: devices would be giving and receiving data each other, working in a network of networks where the information is extracted from the sensors raw data. The IoT has the potential to change the world, just as the Internet did. If we had computers that knew everything there was to know things using data they gathered without any help from us we would be able to track and count everything, and greatly reduce waste, loss and cost. We would know when things needed replacing, repairing or recalling, and whether they were fresh or past their best. These facts are requiring an urgent redefinition of the smart sensor concept, at least in the field of standardization. Every day, smart sensors become smarter in the sense that there are more signal processing resources inside them [2]. Since the "intelligence" has emigrated toward the point of measurement, sensors can do much more than we usually have required from them; in order to obtain not simple samples, raw data, but directly meaningful and useful information on the acquired signal. In order to make the networks interchange more meaningful information or knowledge, a lot of work has to be done for smart sensors. Things can be easily identified into the networks. The protocols that control the traffic over the nets have been defined and established. What is not straightforward to define is the signal information that a transducer should report, especially parameters related with its own signal evolution.

As an example, let's consider a video camera. With signal processing, it will be the most complex sensor that humans have created. It detects people, faces, gestures, shapes, objects, scene description, text, weather conditions, movement, velocity and any pattern on the image. If these signal processing algorithms are executed into the camera, there will flow on the network *knowledge* at low bit rates instead of millions of bit per second without meaning. The lowest data rate is obtained through interrogations with yes/no answers to questions that require signal processing. Note that this approach is better than to compress the signal because it could be compressing a signal that has no meaningful information.

What kind of knowledge should be included into the standard proposal? As a first approximation should be basic information of sensor signal shape, impulsive noise, noise, mean, tendency and detection of patterns. The aim is to include the same information that could provide a human being looking at the signal i.e. to be able to describe the signal and must be of interest for any signal sensor that has been digitized.

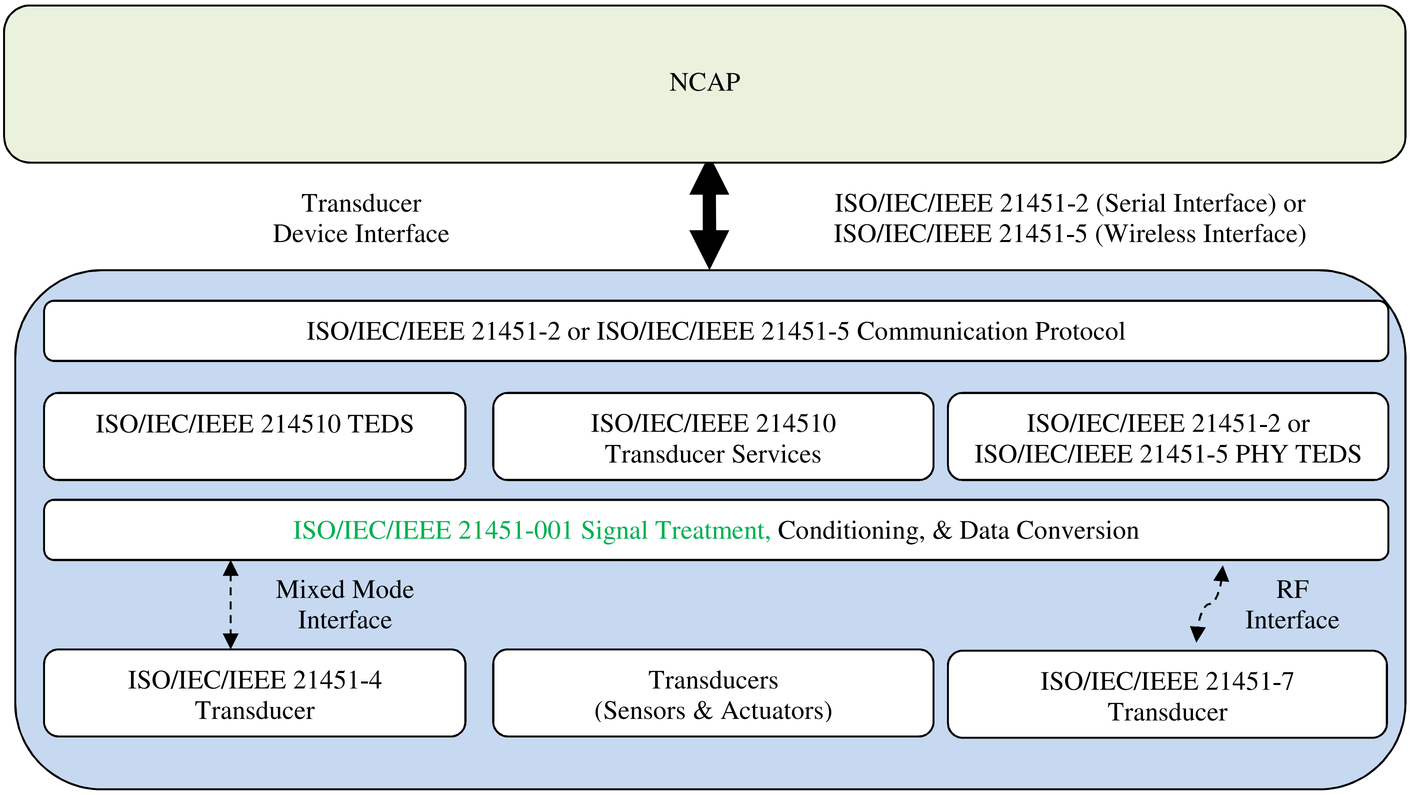

Fig. 1 Block Diagram of the proposal IEEE 21451 standard. (In green the part under developing by the IEEE Standards P21451-001 Working Group)

The following sections present an outline of the proposal: it will be integrated in a new version of the standard IEEE 1451 for smart sensors. In the Fig.1 a block diagram of the proposal standard is shown. Section II presents the algorithm description with emphasis in universal treatment to be applied to any sensor signal. These algorithms can be seen as a smart sampling process. The segmentation and labeling process will be described and it conforms raw information to infer global behavior and to extract important features of the signal. Section III presents the structure of the working group in charge of development this standard.

II. Signal Treatment Algorithms

A. Segmentation

The key process is the signal sampling. This process should be a consequence of the signal characteristics. A time domain algorithm is proposed as a building block for signal analysis. It is based on signal segmentation following of a labeling process. As a result, the smart transducer realizes what kind of signal it is measuring or generating from the sampling process. Computational cost of the proposed algorithms is low so they can be built into smart sensors and executed in real time to provide a normalized platform for sensor signal identification and sensory data cooperation.

It is possible to sample the signal in many different ways, for example the compressive sampling is one of them [8]. The uniform sampling is the simplest way to digitize an analog signal. A strong point is that the time information is captured by the index of the digital data. Where is the information in an analog band limited signal? One way to get this embedded information is to observe the signal trajectory and compare it with known patterns. If we look at the signal in Fig. 2, we do not see isolated sampled, instead how the samples are related each other.

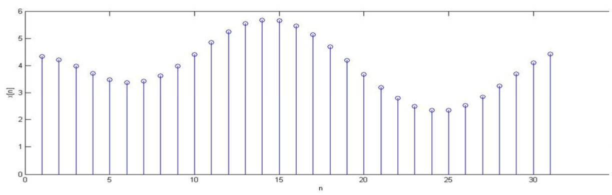

Fig. 2. The uniform sampling gives us samples, but when we observe the signal we describe it on how samples relate each other.

It is easy for the human observer to detect the minimum and maximum, to realize that the transition is smooth and to infer signal shape parameters.

The sampling technique presented emulates a human observer in the sense that relates samples to infer global behavior. The key process is to compare the signal trajectory against linear trajectory. It is not necessary to do this for every pair of samples. Therefore, the first process is to segment the signal. Considering, at first glance, that a sample does not carry important information, it can be replaced by a linear interpolated sample. This operation generates an error, i.e., the

difference between the real and the interpolated sample. By computing the error for the next samples, the non-interpolated ones are tagged as essential when the error exceeds a threshold. Essential samples cannot be obtained from linear combination, therefore in some sense they carry more information. These samples are the boundaries of the segments. Let $x(t)$ the analog sensor signal band limited to **B** hertz and sampled at a frequency of **Fs** with uniform sampling, where **Fs** is much greater than **2B**. Since the signal is oversampled, there will be redundancy between adjacent samples. The algorithm starts interpolating the sample between two samples, i.e. interpolates sample 2 from 1 and 3. Then, the error, as the difference between the real sample and the interpolated sample, is computed. If the error is less than an interpolation error, the sample has no relative importance because its value can be obtained from its neighbors. Now, the sample 3 is interpolated from 1 and 5 and so on. If the error exceeds a threshold, the sample is tagged as important and the process starts again, now from this sample. Interpolation is achieved by linear interpolation. If no important sample is found during **N** samples, the **N**th sample is tagged and the process starts again from this sample. Tagged samples will be the left and right samples of the segment.
As a result of the segmentation algorithm, two vectors are obtained, the tagged or essential samples vector ***mark(n)***, and the temporal index vector ***tempos(n)***. Only a division by two is involved in the average computation. A more complete description of the algorithm is in [3]-[5].

B. *Labeling*

Each time the difference is computed in the interpolation process, the sign of the error is stored. When the segment ends, based on the majority of error signs accumulation, the behavior inside the segment can be classified. The error sign is the clue to classify the segment. If the signal is oversampled, the trajectory of the real signal inside the segment is restricted, by the interpolation error, to four subspaces shown in Fig. 3. These trajectories are classified as **d, e, f** and **g**.

If the segment reaches **N** samples without exceed the interpolation error, classes **a**, **b** or **c** are generated. Fig. 4 summarizes the simplified trajectories.

A new vector, ***class(n),*** is added to the sensor signal description, whose components are the segment class **a, b, c, d, e, f, g,** or **h**. The set of vectors ***mark(n), class(n) and tempos(n)*** in short **MCT,** conform the fundamental structure from which arises all the proposed algorithms. The **MCT** vectors can be seen as a smart sampling process where the relationship among essential samples is clearly stated by the class vector. Why it is necessary an oversampling frequency? In order to get a uniform behavior within the segment, i.e., most of the interpolation errors are all negative or they are all positive, an oversampling frequency is necessary. Under conditions of oversampling and interpolation error approaching to zero, the probability of occurrence of segments **a, b, c** and **h** tends to zero. Therefore, only four classes of segments are important: **d, e, f** and **g.** *The sequence of them identifies the sensor signal.*

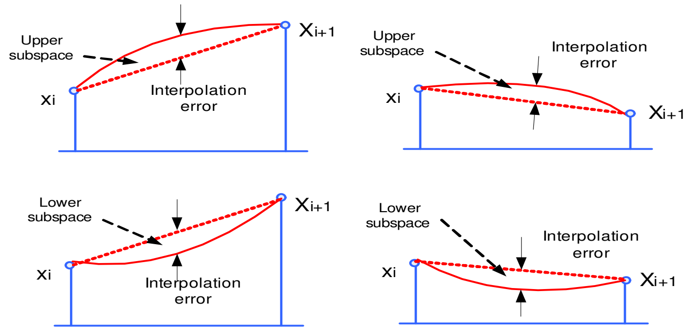

Fig. 3. Depending on the values of extreme samples and how signal departs from linear behavior, there are four possible subspaces for the trajectory of an oversampled signal between tagged samples.

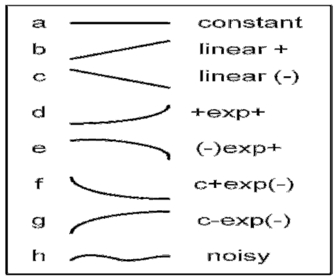

Fig. 4. Class segments are based on the value of the left and right samples and the interpolation error. In classes a, b and c, the segment ends due to it has reaches N samples.

The vector ***class(n)*** captures only the shape information. For example, the pattern "**gefd**" captures a sinusoidal waveform. The frequency and amplitude can be obtained from the **MCT** vectors. Also the ***class(n)*** vector provides precise localization of the signal local maximums and minimums. Maximums occur at the union of "**df**", "**de**", "**gf**" and "**ge**" segments. Minimums occur at the union of "**eg**", "**ed**", "**fg**" and "**fd**" segments. Note that these maxs and mins are controlled by the interpolation error.

III. Outline of the Proposed Standard

A. *Structure of the proposed standard*

Fig. 5 presents a diagram of the proposed standard. The proposal is based on the **MCT** vectors. There is a need to synchronize these vectors with those from other transducers. To analyze data from different transducers a synchronization [9]-[11] algorithm has to be used. At this stage a simple synchronization algorithm has been chosen. A time stamp, e.g. milliseconds from midnight with a circular structure could be a solution. These vectors are the basis for all algorithms as it is shown in Fig. 6. The basic algorithms that can be included in the standard, as a first layer, are exponential, noise, impulsive noise, sinusoidal patterns, tendency and more reliable value. They, in turn, are the basis for more complex algorithms, second layer, which can be optionally included in this standard. A desirable feature would be to be able to send the **MCT** vectors to the system. With this feature, algorithms can be tested outside of the transducer and then embedded in. It is important to note that the **MCT** vectors allow reconstructing the sensor signal. Since it is known the simplified trajectories between essential samples, generating functions can be adapted to approximate the original signal. The simplest generating function is a straight line between the essential samples.

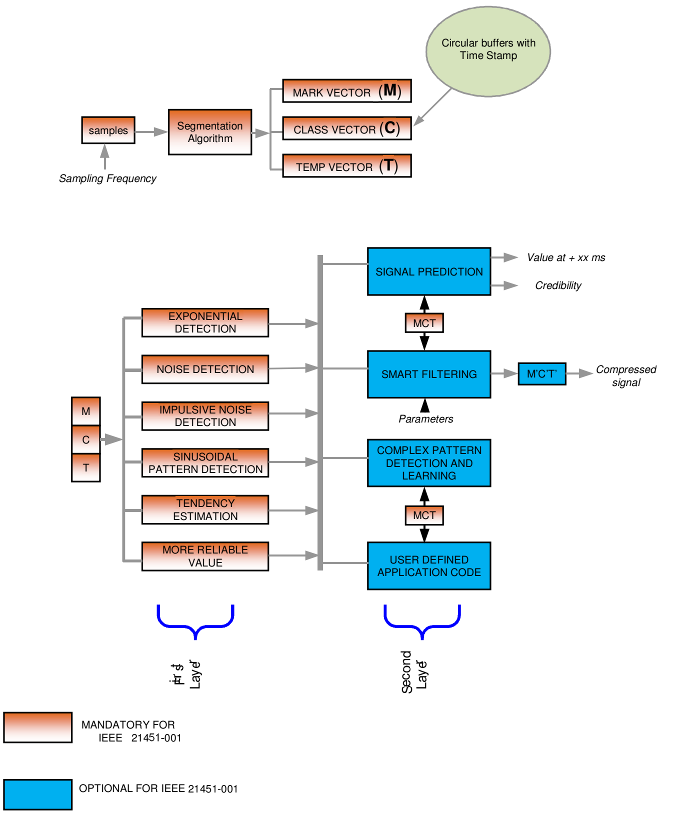

Fig. 5. Block Diagram of the proposed standard. Two layers are proposed.

B. Standard working group organization

The standard working group has been divided into five subgroups to organize the huge task of normalizing sensor signals to be attractive to any kind of sensor.
The objectives are:

Subgroup 1: *To define, propose, and validate first layer algorithms based on MCT vectors. Review the initial proposal, modify it and suggest changes.*

The main challenge in this group is how to organize the **MCT** vectors, i.e. memory assignment, buffer size and time stamp, in order to be input for first and second layer algorithms.
Besides, the group has to define the data structure for first layer algorithms. Observe in Fig. 5 that **MCT** vectors are the main input for every algorithms, therefore the work inside this subgroup is essential.

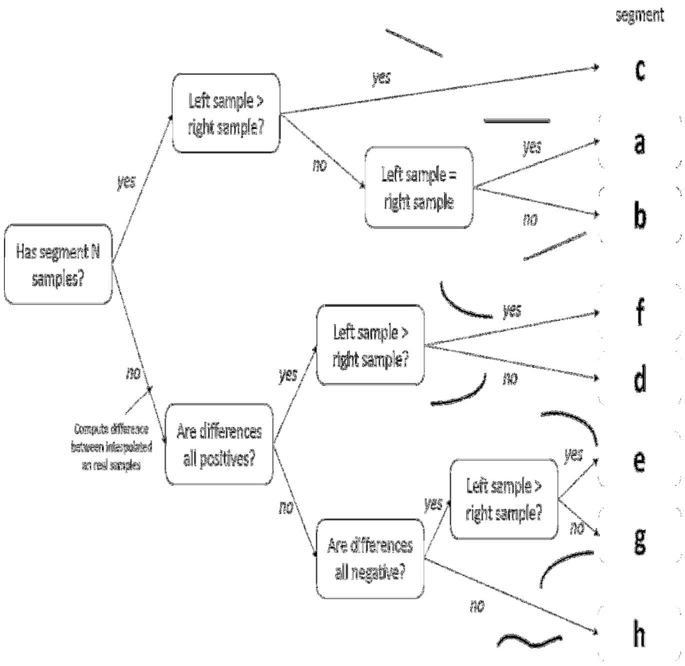

Fig. 6. Block Diagram of the Labeling Algorithm.

Subgroup 2: *To define and coordinate commands and data structure consistently with IEEE 1451 standards. Review and propose a time synchronization scheme for data. Normalize name of variables and data.*

This sensor signal treatment has to be inserted into an existing standard. The IEEE 1451 standard family is mature and still active. This group should consider memory requirements, data and CPU of this proposal with the existing structure of the standard IEEE1451. The signal treatment algorithms of layer one and two need to be parameterized from outside the transducer and data must be sent to / from them. This requires defining commands in concordance with the IEEE1451.

Subgroup 3: *To define, review and propose algorithms based on MCT vectors and first layer outputs for filtering, signal compression and prediction.*

The upper two blocks in layer two, Fig 5; *Signal prediction* and *Smart filtering* are the targets of this group. Since there is information about the shape of the signal, it is possible to predict signal behavior. For example, consecutive segments class "**g**" or "**f**" indicates that the signal tends to a stable value that can be predicted. Also, if has been detected dumped oscillations through "**gefd**" patterns, the long-term value can be predicted.

The MCT vectors describe completely the signal, i.e. from them it can be reconstructed. Linear interpolation between tagged samples is the simplest way to get the signal back, but there are other possibilities for reconstructing.

Since that the original sampled signal lies in an interpolation-error controlled subspace, there is no possibility for aliasing. The reconstructed signal can be seen as a combination of a scaled and shifted version of two generating function:

$$x(t) = \sum a_k \varphi(t/a_k - i) \qquad (1)$$

Where $\varphi(t)$ is the normalized generating function for each subspace as shown in Fig. 7. When linear interpolation is used to rebuild the signal, the two generating functions converge to a straight line between essential samples.

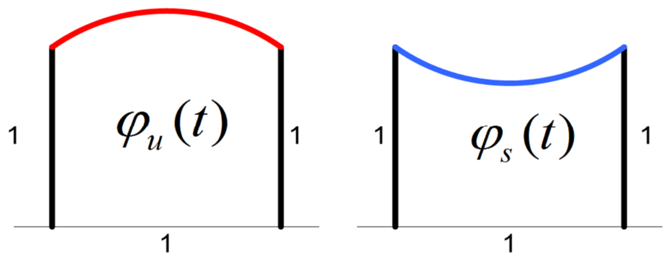

Fig. 7- Normalized generating functions for segments.

Since the function passes exactly through the essential samples, there is continuity between segments and the first derivative coincides, Spline polynomials can also be used to reconstruct the signal within the segment.

If the signal is reconstructed and the sampling process is applied again a new set of MCT vectors is obtained. These M'C'T' could be a compressed version or a filtered version depending upon the input parameters and the number of iterations.

As an example suppose that a sinusoidal signal contaminated with additive Gaussian random noise has to be detected, i.e. to estimate period and phase. From the MCT algorithm it is easy to estimate the period computing the time difference between local maximum. If the period is not the expected, the signal is rebuilt from MCT and MCT algorithm is applied again. Fig.8 shows experimental results after 10 iterations. It is important to remark that from the new set of MCT vectors the period and phase can be easily computed.

Why smart filtering? Because, since there is knowledge about the signal behavior, the signal is filtered when is required. In the example of Fig. 8 the number of iterations could be controlled by the expected sinusoid period.

Subgroup 4: *To propose new algorithms that the group considers that they must be included. Define the data structure for embedding user defined application code and pattern learning algorithms.*

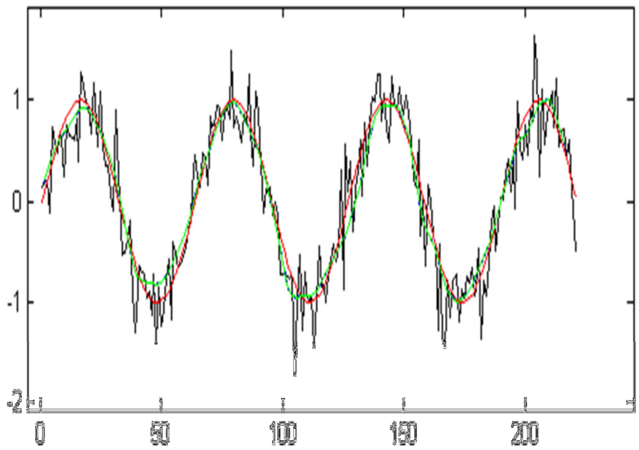

Figure 8- Sinusoid detection with noise. FS=22050, f=345 Hz SNR=7dB. Iterations=10. Interpolation error: 1%. Red: original signal. Green: signal reconstructed from MCT vectors.

The lower two blocks in layer two, Fig 5; *Complex pattern detection and learning* and *User defined application code* are the targets of this group. Layer two algorithms have access to MCT vectors and the outputs of layer one, and they are the building block to extract knowledge. As an example of pattern detection, the detection of QRS patterns in ECG signals was tested with excellent results [12].

The challenge in this subgroup is to define data structure to hold high level signal description that can be embedded into the transducer.

Subgroup 5: *To test and validate proposed algorithms using simulation software (MatLab, Octave, Scilab…) and C code for microcontrollers.*

This group has written code for 8 bit and 32 bits microcontrollers for MCT algorithm. The proposed algorithm in each subgroup has to be tested and validated in this subgroup and this software will be part of the standard.

The subgroups have started to work; each one of them has a leader that reports to the group coordinator. An IEEE central desktop is used by members to share noteworthy documents and, periodically, web meeting take place on WebEx to allow communication, coordination and collaboration between members of the same subgroup and different subgroups.

The main challenge for this standard is that there is no previous work for describing a sensor signal in a universal way.

IV. Conclusions

We live in an interconnected world. Standards will be defined in the IoT in order to identify things and share meaningful information. Data flow will be unmanageable unless only processed information or knowledge being shared.

Sensors are an important subset of things. In the near future, the sensor signal will be processed entirely in the point of acquisition. Therefore, in order to share information, there is a need for a standardized platform. Data grow faster than channels capacity, so the traffic over the network will be around of knowledge, extracted directly from the smart signal sampling process.

The IEEE 1451 is a family of standards for connecting smart transducers to networks [6]. This proposed sampling scheme has been approved as a “Recommended Practice for Signal Treatment Applied to Smart Transducer: P21451-001”. It is under development by the IEEE Standards P21451-001 Working Group.

REFERENCES

[1] Dave Evans, “The Internet of Things: How the Next Evolution of the Internet Is Changing Everything”, CISCO , white paper, April 2011.

[2] Peizhao Hu; Robinson, R.; Indulska, J.*”Sensor Standards: Overview and Experiences”*. 3rd International Conference on Intelligent Sensors, Sensor Networks and Information, 2007. ISSNIP 2007.

[3] G. Monte, “Sensor Signal Preprocessing Techniques for Analysis and Prediction” Industrial Electronics, 2008. IECON 2008. 34th Annual Conference of IEEE. pp 1788-1793. ISBN 978-114244-1766-7.

[4] G. Monte, “A proposal of a New Standard For Sensor Signal Analysis”. Industrial Technology (ICIT), 2010. Chile. IEEE International Conference. pp 1617-1622. ISBN 978-1-4244-5697-0/10.

[5] G. Monte, V. Huang, P. Liscovsky, D. Marasco, A. Agnello,“Standard of Things, First Step: Understanding and Normalizing Sensor Signals”, IECON 2013. Vienna. IEEE International Conference. pp. 116-121.

[6] R. Wynn. “ The TEDS smart sensor revolution”. *IEEE Computing and Control Engineering*. August/September 2004. vol 15, no. 4, pp 25-27.

[7] Nonuniform Sampling, Theory and Pratice. Ed. F. Marvasti, Kluwer Academic/Plenum Publishers, New York, 2000.

[8] E. J. Candès, M. B. Wakin. “An Introduction to Compressive Sampling”. IEEE Signal Processing Magazine. pp. 21-30. 2008.

[9] L. Ferrigno, V. Paciello, A. Pietrosanto, " Experimental Characterization of Synchronization Protocols for Wireless Networks " Proceedings of Virtual Environments, Human-Computer Interfaces, and Measurement Systems (VECIMS 2008), Istanbul, Turkey, 14-16 July 2008.

[10] L. Ferrigno, V. Paciello, A. Pietrosanto, “Experimental characterization of synchronization protocols for instrument wireless interface”, *IEEE Transactions on Instrumentation and Measurement*, vol. 60, no. 3, pp. 1037-1046, 2011.

[11] D. Capriglione, L. Ferrigno,V. Paciello, A. Pietrosanto, A. Vaccaro, “On the Performance of Consensus Protocols for Decentralized Smart Grid Metering in Presence of Measurement Uncertainty” Proceedings of the IEEE International Instrumentation and Measurement Technology Conference (I2MTC) 2013, pp. 1176-1181, May 6-9 2013, Minneapolis (USA).

[12] Monte G. et all. “A novel time-domain signal processing algorithm for real time ventricular fibrillation detection” Journal of Physics: Conference Series Volume 332 2011 J. Phys.: Conf. Ser. 332 012015 doi:10.1088/1742-6596/332/1/012015, 2011.