

Article

Transducer Electronic Data Sheets: Anywhere, Anytime, Anyway

Vítor Viegas ^{1,2,*} , Octavian Postolache ^{2,3} and J.M. Dias Pereira ^{2,4} 

¹ CINA V–Escola Naval, Base Naval de Lisboa, Alfeite, 2810-001 Almada, Portugal

² Instituto de Telecomunicações, 1649-026 Lisboa, Portugal; octavian.adrian.postolache@iscte.pt (O.P.); dias.pereira@estsetubal.ips.pt (J.M.D.P.)

³ ISCTE–Instituto Universitário de Lisboa, 2910-761 Lisboa, Portugal

⁴ ESTSetúbal, Instituto Politécnico de Setúbal, 3810-193 Setúbal, Portugal

* Correspondence: vviegas2@gmail.com; Tel.: +351-925-211-562

Received: 29 September 2019; Accepted: 13 November 2019; Published: 14 November 2019



Abstract: Transducer electronic data sheets (TEDS) are a key element of smart transducers because they support core features such as plug and play, self-calibration, and self-diagnostics. The ISO/IEC/IEEE 21451-4 standard defines templates to describe the most common types of transducers and suggests the use of one-wire memories to store the corresponding data. In this paper we explore new ways to store and access TEDS tables, including near field communication (NFC) tags and QR codes. We also present a mobile TEDS parser, compatible with Android, that is capable of reading TEDS data from all supported mediums (one-wire memories, NFC tags, and QR codes) and decoding them as human-readable text. The idea is to make TEDS available in the easiest way possible. We also underline the need to extend the 21451-4 standard by adding support for frequency–time sensors. A new TEDS template is proposed, and filling examples are presented. The main novelties of the paper are (i) the proposal of new ways to store 21451-4 TEDS tables using NFC tags and QR codes; (ii) the release of new tools to access TEDS tables including a mobile parser; and (iii) the definition of a new TEDS template for frequency–time sensors.

Keywords: TEDS; smart transducer; one-wire; NFC; QR code; frequency–time

1. Introduction

Transducer electronic data sheets—or TEDS for short—are what we get when we convert transducer specifications to digital data and then store them in some kind of electronic memory [1]. Compared to traditional (paper) datasheets, the advantages of TEDS are obvious, namely:

1. TEDS are more sustainable because they are dematerialized and do not waste paper.
2. TEDS are more adaptable as long as they are writable. This allows the storage of dynamic data such as calibration tables and maintenance logs.
3. TEDS are more traceable because they can be embedded on the transducer, thus creating a one-to-one relationship. For new devices, this requires including the TEDS into the manufacturing process; for legacy devices, this requires making modifications to the existing transducer, such as putting the TEDS inside the transducer case or inside the transducer plug.

The TEDS are a key element to accomplish the principle of “transducer plug and play” [2–4]. A microprocessor can read the information stored in the TEDS and use it to configure the measurement chain with minimal human interaction. This process of self-identification and self-configuration reduces configuration errors and improves the interoperability among devices.

The TEDS can also store calibration tables that map the electrical quantity to the physical measurand. These tables are updated with the results obtained from calibration procedures, which can be executed manually by trained personal in the laboratory, or automatically by a microprocessor in situ. In this case, the transducer is said to support self-calibration [5].

Summing up, TEDS are a key element to support advanced features such as self-identification, self-configuration, and self-calibration, which are distinctive marks of smart transducers [6–8].

The ISO/IEC/IEEE 21451-4 standard [9–14] is a good attempt to promote the use of standardized TEDS. It defines data templates to organize transducer specifications and suggests one-wire memories as the preferred medium to store data. The templates cover all the main types of transducers, as well as calibration curves and frequency response tables. The templates are deliberately compact to occupy the smallest amount of memory.

In this paper, we explore new ways to store and access 21451-4 TEDS data to facilitate their use both in the field and in the lab. We also propose a new template to describe frequency–time sensors, which for some reason were forgotten in the 21451-4 standard. Our goal is to disseminate TEDS and, in this way, contribute to a larger adoption of smart transducers and to the standardization of their specifications.

The paper is organized as follows: Section 2 gives the insights of 21451-4 standard; Section 3 reports the most significant work related to TEDS in the last years; Section 4 describes new ways to disseminate TEDS; Section 5 presents experimental results; Section 6 proposes enhancements to the 21451-4 standard; and Section 7 draws conclusions.

2. ISO/IEC/IEEE 21451-4

The ISO/IEC/IEEE 21451-4 [9–14] defines digital wiring, communication protocols, and data templates to make analog transducers smarter (i.e., compatible with the 21451 family of standards [15,16]). The idea is to store the TEDS in a non-volatile memory and access them through a digital communication interface. The digital data flow through the same wiring as the analog signal (multiplexed in time) or use a dedicated one-wire interface as shown in Figure 1. After reading the TEDS, the microprocessor can tune the conditioning circuit for the given transducer (by choosing the best excitation levels, gain values, cutoff frequencies, etc.) so that the analog-to-digital conversion is made optimally. This process can be replicated for multiple transducers because the one-wire interface supports multidrop connections, which is interesting for complex systems with a high number of puzzling transducers.

The TEDS contain all the information needed to handle the transducer properly, including data to identify the transducer, describe its capabilities, and reflect its calibration condition. Nevertheless, all is thought to occupy the smallest amount of memory as demonstrated by the use of compact, dedicated data types such as *Ch5* (five-bit enumeration to represent characters) or *ConRelRes* (logarithmic mapping of an interval).

The TEDS data are organized in three main parts (see Figure 1 again):

1. Basic TEDS (mandatory, read-only, 64 bits length): Contains the transducer ID composed by the manufacturer ID (14 bits), model number (15 bits), version letter (5 bits), version number (6 bits), and serial number (24 bits). The first field (the manufacturer ID) is managed centrally by the IEEE [17], all the other fields are managed by the manufacturer itself. Altogether, the five fields provide a unique identifier.
2. Manufacturer TEDS (optional, variable length): Describes the technical characteristics of the transducer and provides information about its calibration status. The data are organized in pre-defined templates numbered from 25 to 43. Standard templates 25–39 and 43 are used to specify common types of transducers, namely: charge accelerometer/force transducers, voltage output sensors, current-loop transmitters, resistance sensors, bridge sensors, linear/rotary variable differential transformers (LVDT/RVDT), strain gauges, thermocouples, resistance temperature

- detectors (RTD), thermistors, and potentiometric voltage dividers. Calibration templates 40, 41, and 42 are used to characterize the transducer in terms of calibration curve and frequency response.
3. User TEDS (optional, variable length): The user can add free data at the end of the TEDS. These extra data are format free and application specific.

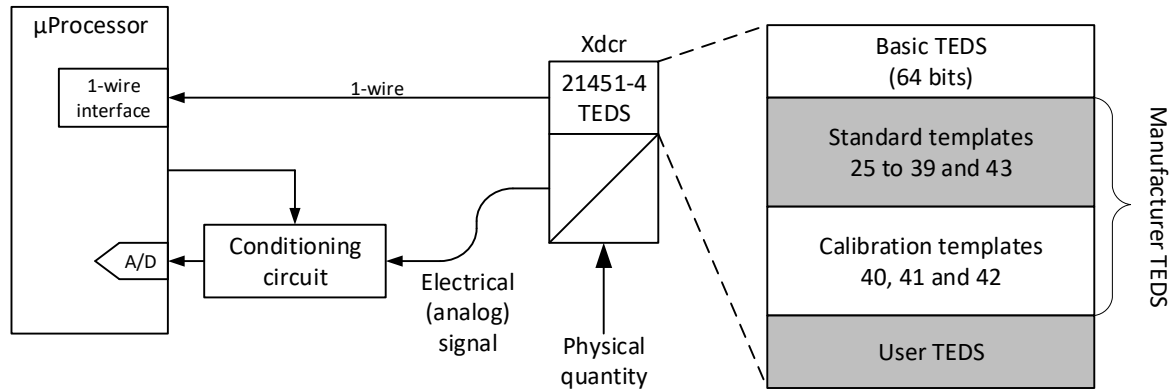


Figure 1. Context of the 21451-4 transducer electronic data sheets (TEDS).

Table 1 shows the use of template 33 to describe a legacy full-bridge load cell, model 505H from Tedeia Huntleigh (a brand owned by the Vishay group). The manufacturer ID has to be defined by the manufacturer because neither Vishay nor Tedeia Huntleigh are presently registered in the IEEE. The lack of information found in many legacy transducers (as is the case) makes it impossible to get a universal identifier (uniqueness is guaranteed inside the organization only).

Table 1. TEDS of a full-bridge load cell.

Section	Parameter	Bits	Value
Basic	Manufacturer ID	14	tbd ¹
	Model number	15	505
	Version letter	5	H
	Version number	6	2
	Serial number	24	tbd ²
Template	Template ID	8	33
	Select case (physical measurand)	6	kgf
	Minimum physical value	32	0
	Maximum physical value	32	+2
	Transducer electrical signal type	—	Bridge sensor
	Select case (full-scale electrical value precision)	2	2
	Minimum electrical output	32	0
	Maximum electrical output	32	+2 mV/V
	Mapping method	—	Linear
	Bridge type	2	Full
	Bridge element impedance	18	350 Ω
	Response time	6	1 s
	Excitation level, nominal	9	10
	Excitation level, minimum	9	0
	Excitation level, maximum	9	15
Calibration date	16	01-06-2019	
Calibration initials	15	VV	
Calibration period	12	365 days	
Measurement location ID	11	21	

¹ To be defined by IEEE. ² To be defined for a given sensor.

The standard template contains fields to describe the technical characteristics of the transducer, including physical units, range, span, bridge type, settling time, and excitation. This information is enough to convert the electrical quantity (Ω) into the physical measurand (kgf) assuming that the transducer is perfectly linear. Details about the real characteristic of the transducer and its frequency response can be added in the calibration TEDS (not present). All standard templates end with basic information about the calibration status (transducer calibrated yes or no? by whom? for how long?).

The TEDS can also be made virtual by saving them on a binary file of type *.ted [18,19]. Virtual TEDS are commonly used in memory-constrained transducers as an extension to the basic TEDS: The measurement system gets the basic TEDS from the transducer, interprets them as a pointer to a web file, downloads the virtual TEDS from the file server, and consumes the information they contain.

3. Related Work

Over the years, the 21451-4 TEDS have been explored to add smartness to the measurement chain. The user TEDS are a good candidate to support transducer and/or application specific features, as shown by Olivieri et al. [20] who stored transducer-specific parameters, not covered by standard templates, in the user TEDS; and by Corotinschi et al. [21] who relied on the user TEDS to extend the functionalities of old measurement equipment. Virtual TEDS are another option to store transducer-related data: in [22] Hernández-Rojas et al. explain how they used virtual TEDS to add plug-and-play capabilities, on the fly, to the nodes of a wireless sensor network.

Other works make use of 21451-0 TEDS, which allow the storage of large, much more complex data structures. It is the case of Morello [23] who stored reference electrocardiogram models in TEDS to improve the diagnosis of heart diseases; or Mitterer et al. [24] who stored security keys in TEDS to authenticate RFID tags used as landmarks for autonomous vehicles; or Ajigboye et al. [25] who used TEDS to support an ontology that facilitates the integration of wearable medical devices into health information systems. The field for extensions is fertile, including using TEDS to balance/optimize the power consumption of dense wireless sensor networks [26].

Another field of extension is the proposal of new templates for transducers that are not covered by the 21451-4 standard. Two good examples are the work of Jevtic et al. [27] who propose a new template for Geiger–Muller detectors; and the work of Kim et al. [28], who propose a new template for electrochemical devices.

In this paper we explore new ways to store and access 21451-4 TEDS, namely by using near field communication (NFC) tags and QR codes, which have the advantage of being contactless and power free. We also provide a full suite of hardware and software tools to access TEDS by all means, including a PC-based data writer to edit TEDS content when the transducer goes to the laboratory for commissioning or calibration, and a mobile parser to read and parse TEDS tables anywhere in the field. This ability to verify TEDS tables in situ can be very useful for all kind of maintenance and asset management tasks, including the commissioning of sensors, actuators, and transmitters in large industrial plants; the verification of sensors in large structures (e.g., bridges and railways); or the inspection of complex machinery containing hundreds of sensors (e.g., aircrafts and pulp-and-paper machines). Our goal is to make TEDS data available anywhere, anytime, the easiest way, as the title suggests.

Last, but not least, we propose a new template to describe frequency–time sensors, an important class of transducers that, in our opinion, was inexplicably forgotten by the 21451-4 standard.

In short, this article makes the following main contributions to the state of the art:

1. Proposal of easier ways to store and access TEDS tables, which do not require wiring, circuitry, or power supply.
2. Presentation of desktop and mobile tools to access TEDS content at the laboratory and in the field.
3. Proposal of a new template to describe frequency–time sensors.

In our opinion, these contributions are innovative because they are useful and new (as far as we know, no other authors have suggested the same). These improvements can, eventually, be included in future revisions of the 21451-4 standard and contribute to a wider acceptance of smart transducers.

4. TEDS Dissemination

This section explores new ways to store and access TEDS tables, including one-wire memories, NFC tags, and QR codes. For all supported mediums, we provide tools to write, read, and parse TEDS content.

4.1. Embedded TEDS

Embedded TEDS refer to traditional TEDS stored in one-wire memories. One wire is a bidirectional, half-duplex, single-master, multi-slave communication interface developed by Dallas Semiconductor. Data are transmitted over a single wire (referenced to the ground), with an active low level and a recessive high level. Bits are usually transmitted at the standard speed of 15.4 kbps (although an overdrive speed of 125 kbps is supported by some devices), and their values depend on the time the signal line rests at the low level (a long time means a 0, a shorter time means a 1). The master is the working bee of the bus because it resets the bus, addresses the target device, issues a command and collects the reply (if any). All data pass through the master. More information about the one-wire bus can be found in [29].

To make an embedded TEDS, we need two basic components: a TEDS editor and a TEDS writer. The TEDS editor is a spreadsheet-like editor where the user can define the structure and the content of the TEDS. The editor shall be able to convert TEDS data from human-readable text to a binary format. The TEDS writer, in turn, takes the TEDS in binary format and stores them in the physical memory handling all low-level details of the one-wire communication interface. Both, the TEDS editor and writer, should be able to perform the inverse operation as well, i.e., read the one-wire memory and convert its contents to human-readable text.

In our case, we have used the TEDS Library for LabVIEW [30] to serve as the TEDS editor for all kinds of standard transducers (templates 25–39 and 43). In the future, we hope to extend the editor to support calibration templates and user-defined TEDS. The editor outputs a virtual TEDS file (with the extension *.ted).

Regarding the TEDS writer, we built a dedicated .NET application, named “TEDS serializer”, that takes the virtual TEDS file and serializes it through a Bluetooth-emulated COM port. At the other end we connect a Bluno board (Arduino UNO with Bluetooth 4.0 [31]) that takes the incoming bytes and writes them into the one-wire memory using the OneWire library [32]. The Bluetooth link is useful because it can be reused by a mobile application (more on that shortly).

Figure 2 shows the data flow from the user to the embedded bit. Figure 3 shows the front end of the serializer, which includes controls to connect to the Bluno board and to select the target memory from the one-wire bus. The “Write” button opens a dialog box to choose the virtual TEDS whose content is dumped into the memory. The bit stream is shown as a hexadecimal string in the text box below. The inverse operation, i.e., read a one-wire memory and save the embedded TEDS as a virtual TEDS is also possible. Tests were made with the chip DS24B33, which is a one-wire EEPROM with 4096 bits, enough to store complex data structures including some space for user TEDS.

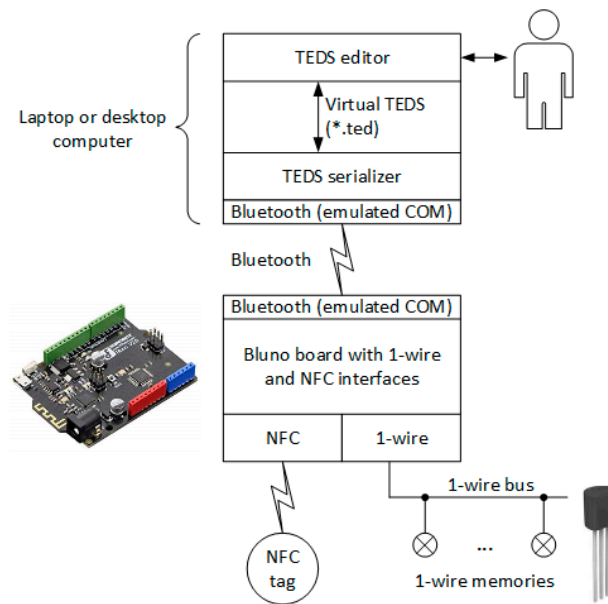


Figure 2. Arrangement to make embedded and near TEDS.

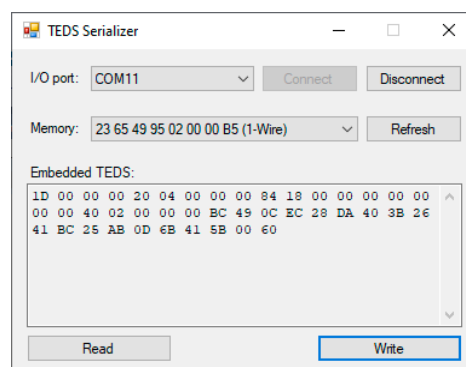


Figure 3. Front end of the TEDS serializer.

4.2. Near TEDS

Near TEDS are like embedded TEDS, with the difference that the memory is accessed through NFC (near field communications). NFC allows the transfer of data between an active device (e.g., smartphone) and a passive device (e.g., memory) through a wireless link that operates at 13.56 MHz. Power is transferred from the active device to the passive device by inductive coupling if both devices are close enough. Proximity is usually seen as an advantage because it provides some protection to human-in-the-middle attacks. The antenna, auxiliary circuitry, and a memory are easily encapsulated into a plastic tag that can easily be attached to any surface. NFC is used today in contactless payments, electronic identity cards, and file sharing. More information about NFC can be found in [33].

As shown in Figure 2, we added an NFC interface to the Bluno board, shield PN532 NFC/RFID controller from Adafruit [34]. The TEDS serializer and the Bluno firmware were upgraded to access NFC tags in addition to one-wire memories. Tests were made with the tag NTAG215 from NXP Semiconductors, with a capacity of 504 bytes = 4032 bits, similar to that of the DS24B33 memory.

4.3. Printed TEDS

Printed TEDS refer to TEDS printed as QR codes as shown in Figure 4. QR codes are matrices of white and black squares, called “modules”, that represent bits of data (0 for white and 1 for black modules). The matrix size can vary from 21 × 21 modules to 177 × 177 modules, in steps of four modules in each axis, each step representing a “version”. The matrix must be surrounded by a white

guard with at least four modules' width. Among the several encoding options available, the most interesting is the "byte" option, which encodes raw binary data according to the ISO 8859-1 standard. Scanners can recover from multiple errors using the Reed–Solomon correction algorithm because the matrix can be filled up to four levels of redundant data (L = 7%, M = 15%, Q = 25%, and H = 30%). Higher correction levels allow data to be recovered more easily in case the module is damaged or obscured. More information about QR codes can be found in [35].

In [36], the authors explain the concept of "printed TEDS" as a way to store TEDS tables in QR codes. Printed TEDS are a very cheap and easy way to disseminate transducer specifications since the codes can be placed on any surface (paper, e-paper, or the transducer case) and can easily be decoded by image scanners without having to power the transducer or connect it to any electronic circuit. QR codes are used today in all kind of tracking applications due to their low cost, robustness, and ease of use.

To create printed TEDS, we built a .NET application, named "QR encoder", that takes a virtual TEDS and generates the corresponding QR code using the ZXing.Net library [37]. As shown in Figure 4, the encoder allows adjustments for image size, margin, and error correction level. The code version is chosen automatically to better accommodate the payload, which is encoded as an array of bytes. The output is a simple PNG image.

To decode a printed TEDS, we need to take a picture of the QR code (using a webcam or equivalent) and convert the image to the corresponding virtual TEDS. In [36], we made the conversion using a commercial library, but free solutions exist as well, such as the ZXing.Net (yes, it works in both directions), Quirc [38] or ZBar [39].

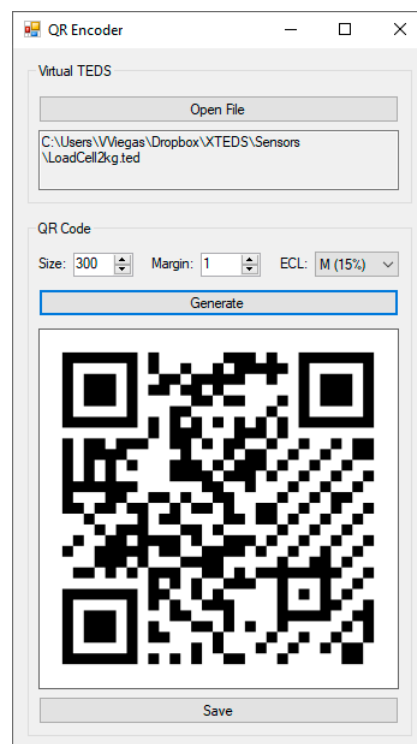


Figure 4. Front end of the QR encoder.

4.4. Mobile Parser

The mobile parser is a very useful tool to read TEDS anywhere, anytime. Our parser was written in C# using Visual Studio 2017 with Xamarin for Android [40] (minimum Android version = 4.1, API level 16—Jelly Bean; target Android version = 8.1, API level 27—Oreo). The parser runs on devices

equipped with a cam (to capture QR codes), NFC interface (to read near TEDS), and Bluetooth (to read embedded TEDS).

The parser starts by presenting three options to access TEDS data (see Figure 5a):

1. Embedded TEDS: The parser connects to the Bluno-based TEDS reader (using a Bluetooth-emulated COM) and gets a list of connected one-wire memories. For a given memory, the embedded TEDS are serialized, parsed, and presented as human-readable text.
2. Near TEDS: The parser searches for an NFC tag filled with TEDS data. If it succeeds, data are read, parsed, and presented to the user as before.
3. Printed TEDS: The parser opens the cam and waits for the user to take a snapshot of a valid QR code. The image is decoded, and the resulting data are parsed and presented to the user as before.

Figure 5b shows the parser's output after decoding the QR code generated in Figure 4. The user can scroll up and down all data fields and share the virtual TEDS with third-party applications (e.g., email or WhatsApp). This can be useful, for example, when the user is working in the field and needs to analyze the TEDS content later, at the office.

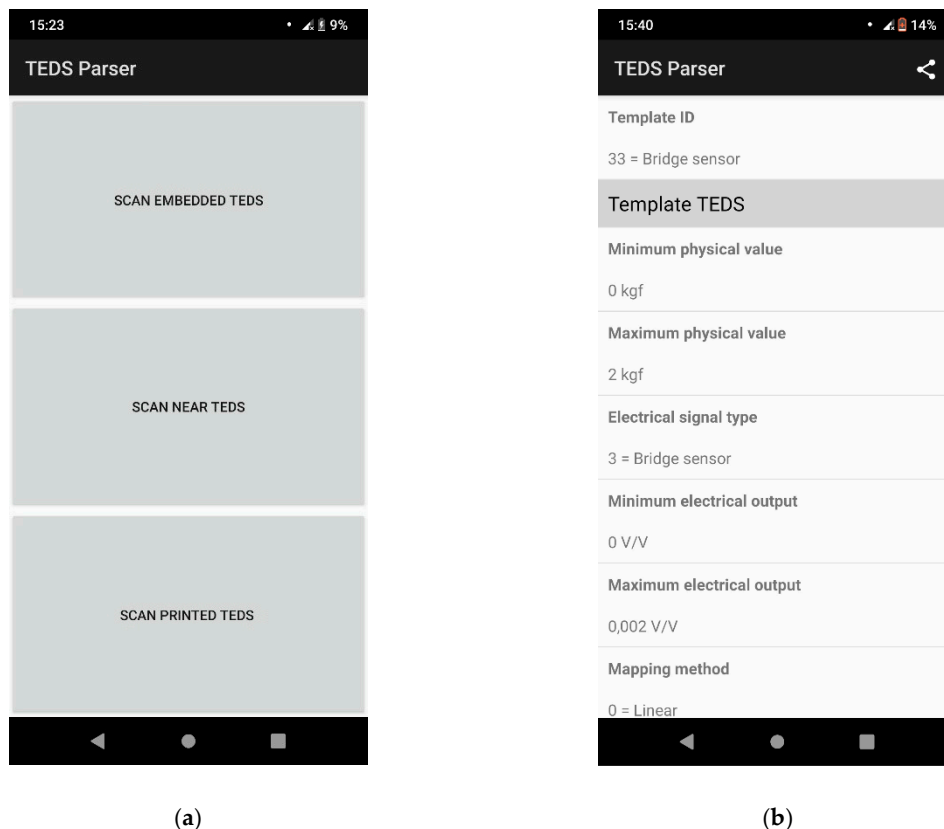


Figure 5. Mobile parser: (a) entry screen; (b) TEDS content presented as a table of values.

5. Experimental Results

We tested our prototypes against commercial equipment compliant with the ISO/IEC/IEEE 21451-4 standard. The equipment was composed by three parts: a universal analog input module NI-9219, a one-slot chassis cDAQ-9171, and the software configurator Measurement and Automation Explorer (MAX), all from National Instruments.

We started by editing the TEDS of a full-bridge load cell, model 505H from Tedeia Huntleigh, as depicted in Table 1. Using the tools described in Section 4.1 (TEDS editor + TEDS serializer + TEDS writer) we converted the TEDS to the bit stream presented in Figure 3 and stored it in a DS24B33 memory. Then, we took the load cell and the memory and connected both to the NI-9219 module, the

load cell attached to the analog front end, and the memory attached to the one-wire interface. Finally, using MAX, we were able to import the embedded TEDS and fill automatically all the fields related with the transducer itself, namely: “signal input range”, “bridge type”, “Vex source”, “Vex value”, “bridge resistance” and scaling values (see Figure 6). In other words, we were able to put in practice the concept of transducer plug and play.

The TEDS of the load cell were also stored in a NTAG215 tag using the NFC interface of the TEDS writer. The tag was then scanned by the mobile parser and presented as human-readable text as shown in Figure 5. The names and values of all fields were parsed correctly, meaning that NFC is viable in storing and retrieving TEDS tables.

The TEDS of the load cell were converted to a QR code using the QR encoder presented in Section 4.3. The output, shown in Figure 4, was scanned by the mobile parser and presented as human-readable text. No errors were detected in this process. Finally, we saved the parsed data back to a virtual TEDS and shared them as an email attachment. Using a binary file comparator, we compared both files (the original virtual TEDS and that received by email) and no differences were detected. In all cases—embedded, near, and printed TEDS—data were stored and retrieved with no corruption. Figure 7 explains, step by step, the experimental procedure.

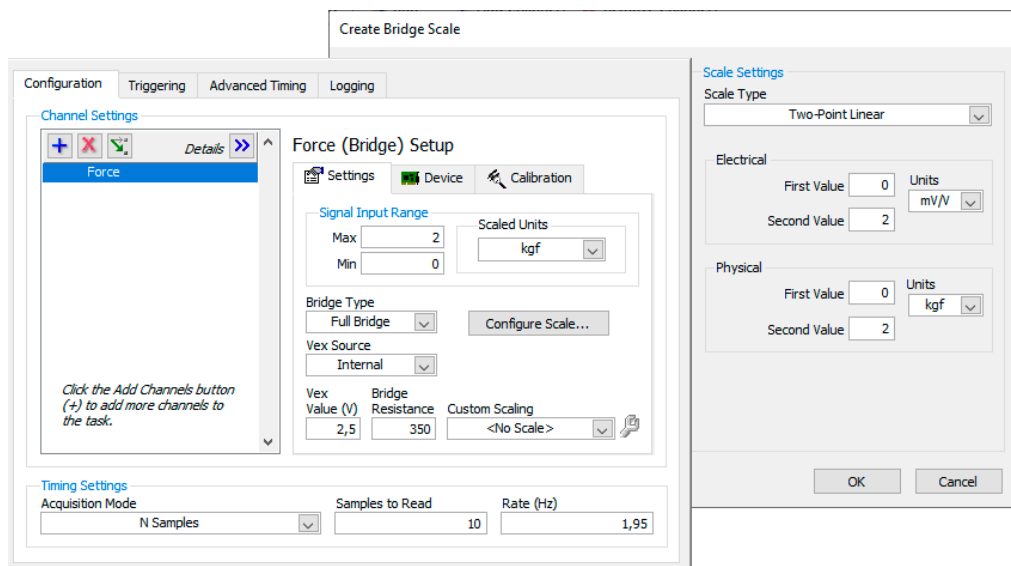


Figure 6. Configuration of the load cell in the Measurement and Automation Explorer (MAX).

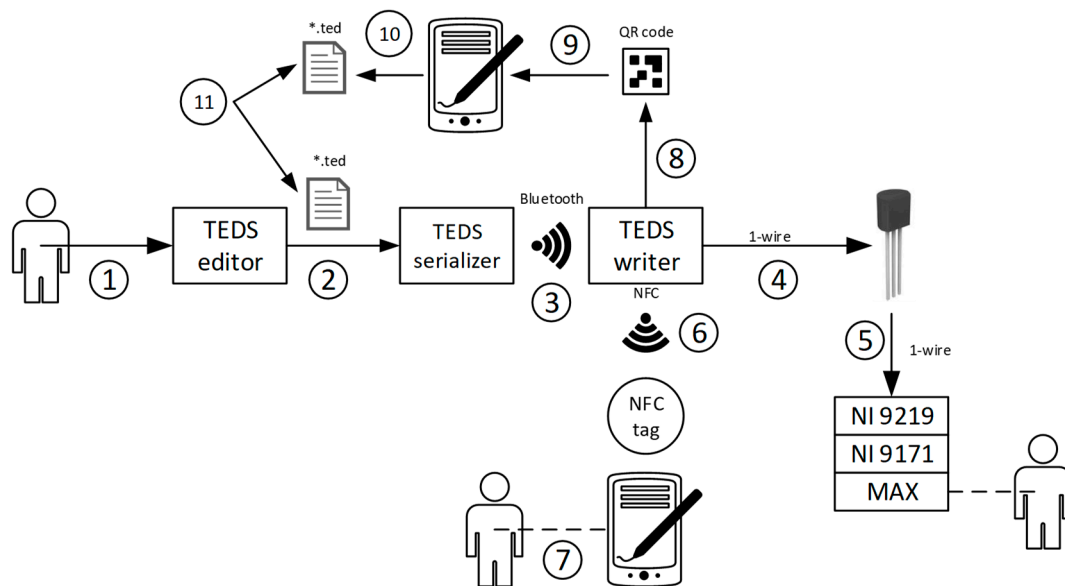


Figure 7. Dataflow through all the tools proposed to disseminate TEDS: (1) using the TEDS editor, the user defines manually the TEDS of the load cell; (2) the TEDS editor generates the corresponding virtual TEDS (binary file *.ted); (3) the TEDS serializer sends the bits of the virtual TEDS, one by one, through the Bluetooth interface to the TEDS writer; (4) the virtual TEDS are embedded into a one-wire memory; (5) the embedded TEDS are read by commercial equipment compatible with the 21451-4 standard, and the content is used to automatically configure the measurement chain; (6) the virtual TEDS can also be stored in a near field communication (NFC) tag; (7) the mobile parser reads the NFC tag and presents the TEDS as human-readable text; (8) the virtual TEDS can also be stored as a QR code; (9) the mobile parser scans the QR code and presents the TEDS as human-readable text; (10) the mobile parser serializes the TEDS to a binary file and sends it by email; (11) the files generated in steps (2) and (10) are equal, bit by bit, thus demonstrating the integrity of the data across the whole apparatus.

A second test was made to evaluate the reading capability of the near TEDS in different conditions of “visibility” (see Table 2). The mobile parser was able to read the NTAG215 tag when it was in line of sight or inside a plastic enclosure (Legrand Plexo IP 55). The reading distance was very short (<2 cm), which aligns with the security concerns of the NFC interface. The tag became invisible (unable to read) when it was covered by aluminum tape (tesa 50525 PV1) or put inside a metallic enclosure. This was expected because the metal surface behaves as a Faraday cage. Therefore, in conclusion, near TEDS can be accessed easily if the tag is not covered by any kind of metallic shield.

Table 2. Reading capability of near TEDS.

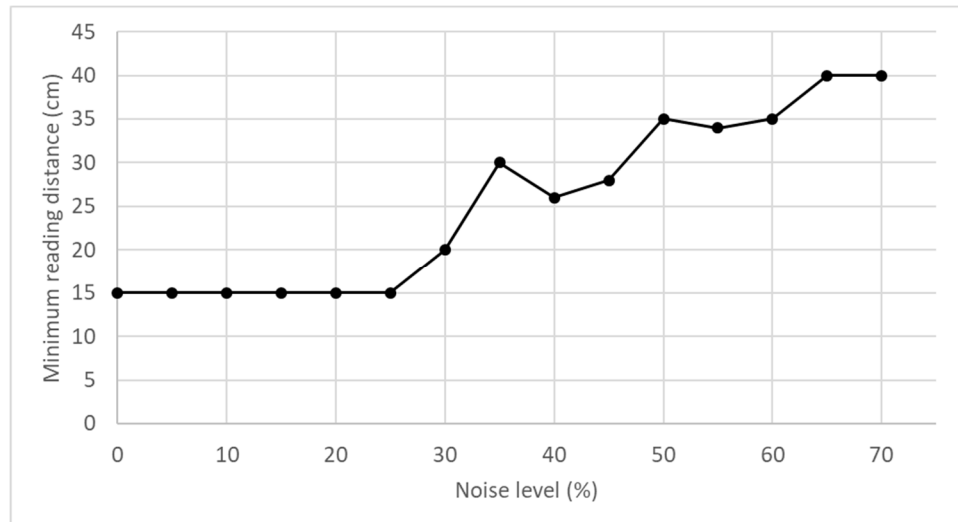
Condition	Read Successful?
NFC tag in line of sight	Yes ¹
NFC tag inside plastic enclosure	Yes ¹
NFC tag covered by aluminum tape	No
NFC tag inside metallic enclosure	No

¹ At a distance <2 cm.

We also evaluated the reading capability of printed TEDS by scanning QR codes corrupted with noise. Taking Figure 4 as a reference, we introduced salt-and-pepper noise in the image, in levels from 5% to 100%, in steps of 5%, and measured the minimum distance at which the scanner could read it. The image was projected on a 14” screen, forming square with 6.3 cm side by side, under typical office light. The results are shown in Figure 8.

The minimum reading distance is about 15 cm, corresponding to the focal distance of the cam. Then we see that, generally, the minimum reading distance increases with the noise level, suggesting

that distance behaves like a low-pass filter that attenuates the effect of the noise. Another interesting feature is the capability of the scanner to read QR codes with very high levels of noise, up to 70%, corresponding to a very distorted image (at least to the naked eye). QR codes with noise level equal or above 75% were unreadable. These results are a good indicator of the robustness of the printed TEDS.



(a)



(b)

Figure 8. Readability of printed TEDS: (a) minimum reading distance versus noise level; (b) some QR codes distorted with salt-and-pepper noise.

6. Enhancements to the ISO/IEC/IEEE 21451-4 Standard

The templates defined by the ISO/IEC/IEEE 21451-4 standard cover many transducer types but not all, as is the case with frequency–time sensors. Moving from the traditional analog signal (voltage or current) to the frequency–time signal (frequency, period, duty cycle, pulse width, phase shift, etc.) offers many benefits, including higher noise immunity, wider dynamic range, and easier signal interfacing (a simple counter can be used to digitize the information) [41]. Frequency–time sensors are an important class of transducers that should be properly handled in future versions of the 21451-4 standard.

Table 3 shows a possible template to describe frequency–time sensors. The proposal tries to meet three constraints: first, follow the structure of existing templates; second, be concise; and third, contain enough information to automatically configure the measurement chain.

The template contains the following sections:

- **Measurement:** gives information about the physical quantity that is being measured and its expected range.
- **Electrical signal output:** describes the electrical signal used as the primary output of the sensor. In the present case, a voltage signal with two amplitude levels is used. Information about the

amplitude levels is useful to detect edges, while information about the frequency range is useful to choose an adequate clock to measure time.

- Time signal output: explains how the timing information is encoded in the electrical signal. Four options are available: time the signal passes in the low level (case 0), time the signal passes in the high level (case 1), period (case 2), and duty cycle (case 3). It is assumed that the measurand maps linearly to the timing quantity.
- Calibration information: provides basic information about the calibration status of the sensor (inherit from existing templates).
- Miscellaneous: provides basic information about the measurement location (inherit from existing templates).

The parameters that affect directly the accuracy of the measurement chain (e.g., duty cycle) are always encoded as ‘Single’ to provide the best resolution. Other, not so critical, fields are encoded with limited resolution to save space (e.g., amplitude levels).

Table 3. Template for frequency–time sensors.

Section	Parameter	Bits	Datatype	Unit	
ID	Template ID	8	Integer	—	
Measurement	Select case-physical measurand	6	Selector	—	
	Minimum physical value	32	Single	Various ¹	
	Maximum physical value	32	Single	Various ¹	
Electrical signal output	Transducer electrical signal type	—	Assume “voltage”	—	
	Low level	9	ConRes (0 to +10.2, step 0.02)	V	
	High level	9	ConRes (0 to +10.2, step 0.02)	V	
	Minimum frequency	7	ConRelRes (1 to 9.48 GHz, ±10%)	Hz	
	Maximum frequency	7	ConRelRes (1 to 9.48 GHz, ±10%)	Hz	
	Response time	6	ConRelRes (1 × 10 ⁻⁶ to 7.9, ±15%)	s	
Time signal output	Select case-signal type	2	Selector	—	
	Case 0 (low time)	Minimum timing value	32	Single	s
		Maximum timing value	32	Single	s
	Case 1 (high time)	Minimum timing value	32	Single	s
		Maximum timing value	32	Single	s
	Case 2 (period)	Minimum timing value	32	Single	s
		Maximum timing value	32	Single	s
	Case 3 (duty cycle)	Minimum timing value	32	Single	s/s
Maximum timing value		32	Single	s/s	
Calibration information	Mapping method	—	Assume “linear”	—	
	Calibration date	16	Date	—	
	Calibration initials	15	Chr5	—	
	Calibration period	12	UInt	days	
Misc.	Measurement location ID	11	UInt	—	

¹ Various units according to Table A.22 of the 21451-4 standard.

Tables 4 and 5 show how to use the new template to describe two commercial sensors: a laser-based distance sensor LIDAR-Lite v3 [42] from Garmin, and a semiconductor temperature sensor MAX6666/7 [43] from Maxim Integrated, respectively.

Table 4. TEDS content for LIDAR-Lite v3 sensors.

Section	Parameter	Bits	Value	
Basic	Manufacturer ID	14	tbd ¹	
	Model number	15	tbd ¹	
	Version letter	5	tbd ¹	
	Version number	6	3	
	Serial number	24	tbd ²	
ID	Template ID	8	tbd ³	
Measurement	Select case-physical measurand	6	m	
	Minimum physical value	32	0	
	Maximum physical value	32	40	
Electrical signal output	Transducer electrical signal type	—	Assume “voltage”	
	Low level	9	0	
	High level	9	5	
	Minimum frequency	7	0	
	Maximum frequency	7	1 kHz	
	Response time	6	10 μ s	
Time signal output	Select case-signal type	2	0	
	Case 0 (low time)	Minimum timing value	32	10 μ s
		Maximum timing value	32	40 ms
	Case 1 (high time)	Minimum timing value	32	—
		Maximum timing value	32	—
	Case 2 (period)	Minimum timing value	32	—
		Maximum timing value	32	—
	Case 3 (duty cycle)	Minimum timing value	32	—
Maximum timing value		32	—	
Calibration information	Mapping method	—	Assume “linear”	
	Calibration date	16	01-06-2019	
	Calibration initials	15	VV	
	Calibration period	12	365 days	
Misc.	Measurement location ID	11	21	

¹ To be defined by the manufacturer (Garmin). ² To be defined for a given sensor. ³ To be defined in future revisions of the 21451-4 standard.

Table 5. TEDS content for MAX666/7 sensors.

Section	Parameter	Bits	Value	
Basic	Manufacturer ID	14	tbd ¹	
	Model number	15	tbd ¹	
	Version letter	5	tbd ¹	
	Version number	6	3	
	Serial number	24	tbd ²	
ID	Template ID	8	tbd ³	
Measurement	Select case-physical measurand	6	°C	
	Minimum physical value	32	−40	
	Maximum physical value	32	+125	
Electrical signal output	Transducer electrical signal type	—	Assume “voltage”	
	Low level	9	0	
	High level	9	5	
	Minimum frequency	7	233.15 Hz ⁴	
	Maximum frequency	7	398.15 Hz ⁴	
	Response time	6	5 s ⁵	
Time signal output	Select case-signal type	2	2	
	Case 0 (low time)	Minimum timing value	32	—
		Maximum timing value	32	—
	Case 1 (high time)	Minimum timing value	32	—
		Maximum timing value	32	—
	Case 2 (period)	Minimum timing value	32	2.512 ms
		Maximum timing value	32	4.289 ms
	Case 3 (duty cycle)	Minimum timing value	32	—
		Maximum timing value	32	—
	Mapping method	—	Assume “linear”	
Calibration information	Calibration date	16	01-06-2019	
	Calibration initials	15	VV	
	Calibration period	12	365 days	
Misc.	Measurement location ID	11	21	

¹ To be defined by the manufacturer (Garmin). ² To be defined for a particular sensor. ³ To be defined in future revisions of the 21451-4 standard. ⁴ Assuming a multiplying factor $m = 1$ by setting pins TS1 = GND and TS2 = V_{DD}. ⁵ Depends on the thermal resistance of the sensor package.

The LIDAR measures distance from 1 cm to 40 m and provides a pulse-width signal with a linear relation of 1 ms/m. The output signal has two voltage levels (0 and +5 V) and is updated on demand by applying a trigger signal. The update rate depends on the measured distance (closer targets allow higher rates) but it can reach up to 1 kS/s.

In turn, the thermometer measures temperature (T) from +40 to +125 °C and converts it to frequency (F) according to the following linear equation: $T(^{\circ}\text{C}) = \frac{F(\text{Hz})}{m} - 273.15$, where m is a multiplying factor defined by two hard-wired pins. Assuming $m = 1$ we have a minimum frequency of 233.15 Hz and a maximum frequency of 398.15 Hz, which corresponds to a maximum period of 4.289 ms and a minimum period of 2.512 ms, respectively.

7. Conclusions

This paper gives alternative solutions to store and access TEDS tables, namely by using NFC tags and QR codes. Both solutions provide an easy interface with smartphones and have inherent advantages that can dictate their choice according to specific application requirements. As main advantages of NFC tags, we can mention good customer acceptance, discretion, and ease of use. In turn, the main advantages of QR codes include visibility, broad consumer knowledge, and wide availability of scanning devices.

A mobile TEDS parser, compatible with Android, that can read and parse TEDS data from one-wire memories, NFC tags, and QR codes is also presented. In these two cases (NFC and QR codes), TEDS is accessed without wiring, circuitry, and power supply, which is very useful for all kind of maintenance and asset management activities, including the commissioning of equipment in large industrial plants, the verification of sensors in large structures, or the inspection of complex machinery. The friendly access to TEDS data, which is the main desideratum of the paper, can contribute to a larger application of smart transducers and to the standardization of their specifications.

The authors also underlined the need to enhance the ISO/IEC/IEEE 21451-4 standard by adding templates to cover more transducer types. A new template, designed specifically for frequency–time sensors, was presented and applied to two commercial sensors.

Author Contributions: Conceptualization, V.V. and J.M.D.P.; methodology, V.V. and J.M.D.P.; software, V.V.; validation, O.P.; resources, V.V., O.P. and J.M.D.P.; writing—original draft preparation, V.V. and J.M.D.P.; writing—review and editing, O.P.; supervision, J.M.D.P.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Dewey, F.R. A Complete Guide to Data Sheets. *Sensors Magazine*. 1998. Available online: <http://www.algrel.com/AllegroPDF/Allegro-1.pdf> (accessed on 6 September 2019).
- Potter, D. Smart plug and play sensors. *IEEE Instrum. Meas. Mag.* **2002**, *5*, 28–30. [CrossRef]
- Boecking, B. Plug-and-Play Sensors Make Big Promises. *ISA InTech Magazine*. 2003. Available online: <https://www.isa.org/standards-and-publications/isa-publications/intech-magazine/2003/may/sensors-on-the-rise-plug-and-play-sensors-make-big-promises/> (accessed on 6 September 2019).
- TEDS. This Is How It Works. Available online: <https://www.hbm.com/en/6258/article-teds-for-cost-savings-short-setup-times-and-application-safety/> (accessed on 6 September 2019).
- Meijer, G.; Pertijs, M.; Makinwa, K. Chapter 2—Calibration and Self-Calibration of Smart Sensors, In *Smart Sensor Systems: Emerging Technologies and Applications*; John Wiley & Sons: Edison, NJ, USA, 2014; ISBN 978-0-470-68600-3.
- Hunter, G.W.; Stetter, J.R.; Hesketh, P.J.; Liu, C.-C. Smart Sensor Systems. *Electromech. Soc. Interface* **2010**, *19*, 29–34. Available online: https://www.electrochem.org/dl/interface/wtr/wtr10/wtr10_p029-034.pdf (accessed on 6 September 2019). [CrossRef]
- Mathas, C. Smart Sensors—Not Only Intelligent, but Adaptable. Available online: <https://www.digikey.com/en/articles/techzone/2011/sep/smart-sensors---not-only-intelligent-but-adaptable> (accessed on 6 September 2019).
- Frank, R. *Understanding Smart Sensors*, 3rd ed.; Artech House: Morwood, MA, USA, 2013; ISBN 978-1-60807-507-2.
- IEEE Standard 1451.4—IEEE Standard for a Smart Transducer Interface for Sensors and Actuators—Mixed-Mode Communication Protocols and Transducer Electronic Data Sheets (TEDS) Formats; IEEE Instrumentation & Measurement Society: New York, NY, USA, 2004; ISBN 0-7381-4007-4.
- Betts, B. Smart sensors—New standard could save lives and money. *IEEE Spectrum*, 2006.
- Mark, J.; Hufnagel, P. What is 1451.4, What Are Its Uses and How Does It Work? Available online: <http://standards.ieee.org/develop/regauth/tut/1451d4.pdf> (accessed on 6 September 2019).
- IEEE 1451.4 Sensor Templates Overview. Available online: <http://www.ni.com/white-paper/3468/en/> (accessed on 6 September 2019).
- Romanchik, D. Tips for Using TEDS Sensors, EDN Network. 2004. Available online: <http://www.edn.com/design/test-and-measurement/4387035/Tips-for-using-TEDS-sensors> (accessed on 6 September 2019).
- Jevtic, N.; Drndarevic, V. Development of smart transducers compliant with the IEEE 1451.4 standard. In Proceedings of the 8th IEEE International Symposium on Instrumentation and Control Technology (ISICT), London, UK, 11–13 July 2012. [CrossRef]
- Song, E.Y.; Lee, K. Understanding IEEE 1451—Networked smart transducer interface standard. *IEEE Instrum. Meas. Mag.* **2008**, *11*, 11–17. [CrossRef]

16. Wobschall, D. Networked sensor monitoring using the universal IEEE 1451 standard. *IEEE Instrum. Meas. Mag.* **2008**, *11*, 18–22.
17. IEEE Standards Association. Available online: <https://standards.ieee.org/products-services/regauth/manid/index.html> (accessed on 6 September 2019).
18. Creating Virtual TEDS. Available online: <http://www.ni.com/example/26498/en/> (accessed on 6 September 2019).
19. Reading Smart TEDS Sensors and Virtual TEDS Files in LabVIEW. Available online: <http://www.ni.com/example/27195/en/> (accessed on 6 September 2019).
20. Ulivieri, N.; Distante, C.; Luca, T.; Rocchi, S.; Siciliano, P. IEEE1451.4: A way to standardize gas sensor. *Sens. Actuators B: Chem.* **2006**, *114*, 141–151. [[CrossRef](#)]
21. Corotinschi, G.; Găitan, V.G. The development of IoT applications using old hardware equipment and virtual TEDS. In Proceedings of the IEEE International Conference on development and application systems (DAS), Suceava, Romania, 19–21 May 2016. [[CrossRef](#)]
22. Hernández-Rojas, D.L.; Fernández-Caramés, T.M.; Fraga-Lamas, P.; Escudero, C.J. A Plug-and-Play Human-Centered Virtual TEDS Architecture for the Web of Things. *Sensors* **2018**, *18*, 2052. Available online: <https://www.mdpi.com/1424-8220/18/7/2052> (accessed on 9 September 2019). [[CrossRef](#)] [[PubMed](#)]
23. Morello, R. Use of TEDS to Improve Performances of Smart Biomedical Sensors and Instrumentation. *IEEE Sens. J.* **2015**, *15*, 2497–2504. [[CrossRef](#)]
24. Mitterer, T.; Gietler, H. Lisa-Marie Faller, Huber Zangl, Artificial Landmarks for Trusted Localization of Autonomous Vehicles Based on Magnetic Sensors. *Sensors* **2019**, *19*, 813. [[CrossRef](#)] [[PubMed](#)]
25. Ajigboye, O.S.; Danas, K. Towards semantics in wearable sensors: The role of transducers electronic data sheets (TEDS) ontology in sensor networks. In Proceedings of the IEEE 18th International Conference on e-Health Networking, Applications and Services (Healthcom), Munich, Germany, 14–16 September 2016. [[CrossRef](#)]
26. Amin, F.; Zubair, M. Energy-efficient clustering scheme for multihop wireless sensor network (ECMS). In Proceedings of the 17th IEEE International Multi Topic Conference 2014, Karachi, Pakistan, 8–10 December 2014. [[CrossRef](#)]
27. Jevtic, N.; Drndarevic, V. Plug and Play Geiger-Muller Detector for Environmental Monitoring. *Instrum. Sci. Technol.* **2014**, *43*, 222–243. [[CrossRef](#)]
28. Kim, J.-D.; Kim, D.-J.; Byun, H.-G.; Ham, Y.-K.; Jung, W.-S.; Han, D.-W.; Park, J.-S.; Lee, H.-L. The definition of basic TEDS of IEEE 1451.4 for sensors for an electronic tongue and the proposal of new template TEDS for electrochemical devices. *Talanta* **2007**, *71*, 1642–1651. [[CrossRef](#)] [[PubMed](#)]
29. Linke, B. Tutorial 1796—Overview of 1-wire technology and its use, maxim integrated. 2008. Available online: <http://pdfserv.maximintegrated.com/en/an/AN1796.pdf> (accessed on 6 September 2019).
30. How Do I Install the TEDS Library into LabVIEW? Available online: <https://knowledge.ni.com/KnowledgeArticleDetails?id=kA00Z0000019LmDSAU&l=pt-PT> (accessed on 10 September 2019).
31. Bluno—An Arduino Bluetooth 4.0 (BLE) Board. Available online: <https://www.dfrobot.com/product-1044.html> (accessed on 6 September 2019).
32. OneWire Library. Available online: https://www.pjrc.com/teensy/td_libs_OneWire.html (accessed on 6 September 2019).
33. What is NFC. Available online: <https://www.shopnfc.com/en/content/9-what-is-nfc> (accessed on 6 September 2019).
34. Adafruit PN532 RFID/NFC Breakout and Shield. Available online: <https://learn.adafruit.com/adafruit-pn532-rfid-nfc/overview> (accessed on 6 September 2019).
35. Answers to Your Questions about the QR Code. Available online: <https://www.qrcode.com/en/> (accessed on 6 September 2019).
36. Viegas, V.; Pereira, J.M.D.; Girão, P.S.; Postolache, O. Transducer Electronic Data Sheets: Why Not Print Them? In Proceedings of the 2014 International Conference and Exposition on Electrical and Power Engineering (EPE), Iasi, Romania, 16–17 October 2014.
37. ZXingNET. Available online: <https://archive.codeplex.com/?p=zxingnet> (accessed on 6 September 2019).
38. Quirc. Available online: <https://github.com/dlbeer/quirc> (accessed on 6 September 2019).
39. ZBar Bar Code Reader. Available online: <http://zbar.sourceforge.net/> (accessed on 6 September 2019).

40. Xamarin.Android. Available online: <https://docs.microsoft.com/en-us/xamarin/android/> (accessed on 6 September 2019).
41. Yurish, S.Y. Sensors and transducers: frequency output versus voltage output. *Sens. Transducers J.* **2004**, *49*, 302–305.
42. LIDAR-Lite v3. Available online: <https://buy.garmin.com/en-US/US/p/557294/pn/010-01722-00> (accessed on 6 September 2019).
43. MAX6667. Available online: <https://datasheets.maximintegrated.com/en/ds/MAX6666-MAX6667.pdf> (accessed on 6 September 2019).



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).