ABSTRACT
It has become ever more necessary to integrate a wide range of different electric devices into a digital information system capable of helping make timely decisions based on the many variables coming from all such devices. This integration usually takes place over a telecommunications network.

The paper addresses this need, especially regarding devices or systems originally designed to provide primary electric power or other services dealing with high power, like generators, motors, air-conditioning systems, and lighting systems. In this context, the term “dumb” refers to the fact that all these devices were not originally designed to communicate with a computer, mainly because many of them came into widespread use years before any computer system was even available. But, for example, present-day specifically designed very low power devices, such as those used in RFID (Radio Frequency IDentification) applications are, strictly speaking, also “dumb” even though extremely useful and ever more present as sources of input for computer-based information systems.

What a professional in Information Technology might be presently lacking, while addressing development needs to interact with the devices mentioned in the previous paragraph, is a technical understanding of the physical variables involved, in such a way as to allow for the best possible use of whatever data the “dumb” devices actually yield, efficiently integrating these data into their programs. Therefore, the paper suggests a software-centered treatment of the variables involved in the process, and explains their physical characteristics in an attempt to allow better hardware integration into an intelligent software environment.

Keywords
Dumb devices; electric signals; intelligent devices; interfacing; hardware integration; signal processing; transducers.

1. INTRODUCTION
While trying to build an interface with any piece of electric equipment other than a computer, software engineers or developers might often feel that they lack the expertise or even specific information on how to interact with non-processing hardware. But by grasping and applying some elementary concepts in electricity and physical signals, a way toward integration can be rather easily followed, bringing the “alien” piece of electric machinery into the technical realm of the software engineer. Even more: a clear understanding of the physical signals actually involved could even allow for integration with some non-electrical devices, offering surprisingly flexible tools to expand software applications.

By properly assessing how far does the application need to go toward a formal definition of “intelligence” in order to achieve a truly intelligent practical solution of the problem lying behind a technical requirement, an extremely useful interaction between a computer and a traditionally “dumb” device could be achieved, allowing the end-user to exploit in full the digital processing resources of both the computer and the communications network available.

2. “INTELLIGENT” AND “DUMB” DEVICES
2.1 An Operational Definition of Intelligence
In 1950 (sixty years ago, no less!), the British mathematician Alan Turing (1912–1954) proposed what is now known as the “Turing Test” as a “replacement for the question Can machines think?” [5].

Basically, the present version of the Turing Test was developed upon what Turing called the Imitation Game in his seminal article “Computing Machinery and Intelligence” [8].

Figure 1 illustrates the principle underlying the Imitation Game: the game is played with a man (M), a woman (W), and an interrogator or juror (J) whose gender is irrelevant to the game. For the sake of simplicity, let us assume however that J is a woman. J is kept in a different room where she has no contact with M or W, except through a teletype writer (remember this was first written in 1950!). We can now think that, as far as J is concerned, the only possible communication with M and W works through a text-only HyperTerminal-like program or interface, but the point is J actually does not know who is the man and who is the woman, and can neither see nor hear them. J’s objective is to determine who is who by questioning both M and W through the interface, while their objective is to deceive...
2.2 Traditionally “Dumb” Devices

J into thinking that the actual situation is the other way around.

The questioning and the deceiving process take place exclusively by means of the interface. Therefore, J will use it to ask her questions by writing them down in whichever natural language all three players might be familiar with; and M and W will also use the interface to answer the questions in this same natural language.

The questions can be on any subject. For example, Turing suggested they be on mathematics, the weather, poetry, and chess, but this is actually inconsequential if the test is to be truly open or unrestricted.

The most important point in Turing’s paper seems to be that, if any entity is capable of imitating or mimicking a conduct different from what belongs in its own nature, it deserves to be labelled “intelligent”.

Therefore, the Turing Test extrapolates the Imitation Game by asking if M or W (any of the human players apart from the juror) could be convincingly replaced by a digital computer.

Figure 2 shows what can be considered as a modern version of the Turing Test. Let us bear in mind that, even though the test has been presented here in a somewhat modified version in order to better fit it into present-day technological concepts, the main point is still there: the digital computer will be deemed “intelligent” if, by using this set-up, human jurors can be deceived into thinking they are actually interchanging information with another human being; or be unable to decide which answers come from the machine, and which come from a human player.

Figure 2: The actual Turing Test.

Another interesting fact: by the time of Turing’s paper, the most advanced digital computers were still very far from performing the test even in a slightly convincing way. This was true of both their hardware and their software, so the Turing Test provides a true and realizable abstraction as far as a feasible operational definition of “intelligence” is concerned.

2.2 Traditionally “Dumb” Devices

The context that interests us here, the Merriam Webster Dictionary defines “dumb” as “Not having the capability to process data.” This is also the most common meaning of the word among members of the Information Technology community at large.

In fact, the Loebner Prize, a gold medal and a check for $US 100,000 will be granted to the first program capable of passing an unrestricted version of the Turing Test. Even though bronze medals and checks for $US 3,000 have been awarded for the past 20 versions of the Loebner contest, the full prize is yet to be collected so, from a formal point of view, not only have computers been interacting exclusively with dumb devices since their very first appearance in the technological arena more than sixty years ago (and continue to do so), but they are still dumb devices themselves!

It is worth mentioning that the Loebner prize has been much criticized regarding its approach to Artificial Intelligence (AI). The well-known Marvin Minsky (The Society of Mind [3]), co-founder of the MIT Artificial Intelligence Laboratory (1959), and presently considered to be the foremost living expert in AI, is perhaps the best-know critic: “In fact, Marvin Minsky has offered $US 100 to the first person who can get Hugh Loebner to revoke the competition, which he [Minsky] calls an ‘obnoxious and unproductive annual publicity campaign’.” Loebner astutely declared Minsky a co-sponsor of the contest, since, according to the rules, when the grand prize is won, the contest will not be held again. In that case, with Minsky’s contribution, the prize should become $US 100,100.” [6].

According to Minsky, this competition is not about AI at all, even though during past instances of the Loebner some judges have indeed mistaken computers for humans and conversely. Even the Turing Test as a whole might not be precisely about “intelligence”, but rather about an ability to play with analogies and words. Do we, humans, offer at all times sensible answers, or even answers that do not vastly digress? Does the fact that sometimes a human player has...
been mistaken for a computer mean that player is not an intelligent being?

This clearly shows that there is presently a wide range of applications and devices capable of communicating flexibly enough to be extremely useful in practice, even though they might not be labeled “intelligent” from a more formal, “strictly Turing” point of view.

Traditionally, non-processing devices were, of course, non-electrical devices plus high-power electrical devices such as transformers, motors, lighting systems; and low-power electrical devices such as locks and non-digital devices in general. This paper also includes an elementary definition of what a digital (and, therefore a “non-digital”) device is, so let us not worry about a more precise definition at this point.

We will see how to benefit from physical aspects ("physical response") of all these devices by integrating their output into a modern data network or, at least, by properly feeding it into a present-day computer.

2.3 Devices that are Presently Perceived as Intelligent

We have already seen that it is rather difficult (and maybe even useless) to apply the Turing Test whenever a practical definition of “intelligence” is required. Let us concentrate instead on a more usual and more useful definition through modern everyday applications in the technological field. The Merriam Webster dictionary will also help us in this purpose: in a context which lies closer to ours, it defines an “intelligent device” as “Guided or controlled by a computer; especially: using a built-in microprocessor for automatic operation....”

The keyword here seems to be “microprocessor” which actually conveys the more general idea of a micro-chip. From a non-technical point of view, we seem prone to accepting a device as capable of intelligence whenever we know or see that those many-legged electronic components are part of it. Also, with the present prevalence of digital over analog technology, it is more likely that the chip be a digital one, truly capable of automatic data processing, rather than an analog component which has been mistaken for an “intelligent” counterpart.

But, taking into account the present state of technological development and the importance of this particular market niche, we must also recognize that RFID tags are perceived as “intelligent” in practice, even though they do not go beyond knowing their own number or “name” and maybe saving some additional information.

Because of their particular importance, we shall later dedicate a specific section to those devices.

3. THE BRIDGE

3.1 Physical Variables and Signals

A signal is usually defined as any physical variable [9]: any physical entity that varies in time and possibly space. In order not to delve too much into the meaning of these words, let us only recall that “physics” comes from the Greek for “nature”. Therefore, we shall call any perceptible change (variation) occurring in nature a “signal”: any change which we are able to detect, perceive, or acknowledge as such.

Actually, in this context, we are using the words “physical variable” and “signal” as synonyms.

3.2 Two Important Electric Signals

It is undeniable that the word “computer” has become the short form for “general purpose electronic digital computer”. Even without attempting a definition of any of these words, and relying again only on our intuition, we can doubtlessly perceive the particularly important role that electricity—and electronics must be certainly related to electricity too—plays in the field of Information Technology.

Therefore, “electric signals”, signals of an electric nature, play a very important role whenever one tries to interact with a computer or just with any chip capable of “remembering” its own name!

We shall rely once more on intuition and not attempt any formal definition: electrical phenomena are the kind of phenomena that makes tiny sparks issue from your clothing by rubbing it in a dry day, or that sometimes gives you an unpleasant start when touching a car door and, certainly, it is the kind of phenomena that we regularly use to convey information from one point to a distant one, even if the distance involved is of many thousands of miles. Of course, it is also the same kind of phenomena that makes automatic data processing a reality.

Having said this, we need only get somewhat familiar with two very important electrical variables: the so called “voltage” and “current”. In order to understand those technical terms, we shall still appeal to the reader’s intuition and patience. Figure 3 attempts to convey this intuition graphically by means of a bicycle pump and air flow.

In order for a “voltage” (also called “electric tension” or “difference in electric potential”) to exist, it must be present between two different points, even if nothing actually “flows” between these points: back to our analogy with the air pump, we could use a finger to block the air at the end of the rubber tube so no air flows out of it but, by pressing down the plunger of the bicycle pump, we would then feel the air pressing against our finger. While talking of electrical phenomena, this pressure will be analogous (similar) to the electric condition needed to initiate and sustain electric flow or current between two different points as soon as there is a conveniently closed path uniting them.

The “current” (“electric current”), on the other hand, will be similar to the air flow. Notice that if there is no closed path, if there is no rubber tube, it will not be possible to establish an air current: upon operating the plunger, the air will just exit from the pump in no particular direction, and we shall not be able to make it flow toward (or into) a desired destination. Even though this analogy fails in several aspects (including the way the path is “closed”), and it is far from being physically correct, it seems to be enough to illustrate the underlying idea in non-technical terms: we can speak of a flow or current only if the corresponding physical entity (be it air or electricity) moves under “pressure”, in a
Because of the very many different forms energy actually takes in the physical world, there are also that many kind of transducers.

4. ANALOG AND DIGITAL ELECTRIC SIGNALS

4.1 Some Definitions

The easiest non-formal way to differentiate between “analog” and “digital” technology is to remember how we have used the word analog (or analogous) as synonymous to similar, and then remember the origin of the word digital: it comes from digit meaning finger, so it can be also rendered as finger-like.

We have an analog variable whenever its form is similar to that of any other signal appearing in nature and, since nature does not allow immediate or even extremely sudden changes, does not allow for broken phenomena2, an analog signal makes a smooth graph when plotted against time: a graph with no vertical gaps; a graph where any intermediate value is possible between any other two.

A digital variable, on the other hand, allows only for discrete or separate values, like the fingers in the hand: using the fingers or digits one can only count in integers, without the possibility of intermediate values, as we do not have say 3.5 fingers available, but just 3 or 4 instead. A digital signal, therefore, when plotted against time, makes a step-like graph with only a set of well-defined values and no possible result in between.

Figure 4 shows the basic difference between an analog and a digital signal: in the former, any intermediate value is possible between any other given two; in the latter, there is only a predefined set of feasible values with no possibility of an intermediate one. In this case, the possible outcomes are just separated individual entities like the fingers in the hand. Notice that the detection of an analog variable can be easily associated with a (continuous) measuring process to establish a similarity between the actual variable and the reading in the measuring instrument, while the detection of a digital variable is rather associated with a (discontinuous or “discrete”) counting process as might occur while using the fingers.

In Information Technology there is, of course, a very important kind of digital signals: binary signals. Since it is easier by far to detect only the absence (“0”) or the presence (“1”) of an electric signal, rather than to measure a specific value within any given range, already the first digital computer was built upon a binary, namely a “two-valued”, numbering system which allows for the physical realization using only the presence or the absence of electrical signals. We can easily (or at least intuitively) understand that it must be simpler to say “there is a current” or “there is no current” rather than saying “there is a current of 0.2 amperes”. This is the rationale behind the characteristic sequences of zeros and ones that we know are at the core of the “native language” of any electronic data processing or telecommunications system.

2At least at the macroscopic level. The statement does not take into account quantum phenomena.
Before leaving this section, it is also worth mentioning that the "digitalization" or limitation of possible outcomes only to fixed values, rather than to any intermediate one, can take place over either the physical variable under consideration (say the electric current) or specifically over time (one might be interested in scanning the signal only at fixed intervals rather than staring at it all the time). Figure 5 illustrates this idea: the digitalization can take place over the vertical axis, the horizontal axis (usually representing time), or over both of them.

4.2 Fourier Techniques: A “Four-year” Commitment?

Jean-Baptiste-Joseph Fourier (1768-1830) was a French mathematician and physicist who concerns us here for his extraordinary insight of physical variables or signals. While dealing with the phenomenon of heat transmission, Fourier proved that any signal can be thought of as being made up of just a sum of sines (or cosines) at different frequencies (the so-called “harmonics”). That is, sines or cosines oscillating or changing at different rates in time. Figure 6 illustrates this concept.

Moreover, any such signal can be fully rebuilt using just a set of samples measured or gotten at least twice as fast as the fastest sine or cosine (twice the “highest harmonic”) in the summation we have referred to.

College students whose native tongue is English usually tell a joke about Fourier techniques, saying that the mathematics involved is so complex, that this undertaking actually implies a “four-year” commitment. Their true complexity is, of course, far below what this label suggests but, even if it were as complex as the students’ nickname says, we do not need to delve into its details because, right now, there is a wide range of commercially available microchips ready to do all the Fourier math for us and deliver a nice binary signal, a nice sequence of 0’s and 1’s, after working very quickly on the analog current or voltage signal produced by some transducer which is also widely available.

5. RADIO FREQUENCY I.D. APPLICATIONS

5.1 What is RFID Technology?

The initial concepts regarding RFID technology were probably born as long ago as 1945: right as World War II was coming to its end, and as the international tension began escalating toward what has been called “The Cold War” [7].

Presently, we only need to know that this technology involves the capacity to yield or give a unique identifying number using radio waves. Nowadays, this can be done simultaneously over a universe of thousands of items: thousands of different items can identify themselves via tags whose embedded information could be read into a data base working on a designated server, by polling or probing the tags using radio waves, that is, without the need of a physical contact between the tag and the reader.

Wal-Mart Stores Inc. doubtlessly gave a powerful push to the then still somewhat incipient RFID industry more than five years ago, by announcing its readiness to require that it be used by its main suppliers [4].

We shall not go into the details of RFID technology here, but we shall only mention that the total market size of RFID technology worldwide was already approaching $US 9,000 million ($US 9 billion) by the end of 2009. This is, of course, clear proof of the importance of RFID technology in the present day world.
5.2 Is an RFID Tag Intelligent?
An RFID tag is only a minuscule, a millimetric, microchip-antenna set capable of storing its own unique identification and delivering it to the pertaining reader over a radio wave. It is so inconspicuous and unobtrusive, that it is frequently used to identify books in a library, or pieces of clothing in a public laundry, or to tag animals as small as butterflies or as big as cows.

It does, of course, not come even close to passing the Turing Test. In spite of this, it is “smart” or “intelligent” enough to know its own “name” and give it upon proper request, actually going “beyond tracking” [1].

Therefore, this application makes a very good example of present-day technology which, even though failing the Turing Test, is in fact “intelligent” enough to be one of the most important factors driving modern Information Technology.

6. DSP: A TRUE “FOUR-YEAR” COMMITMENT
Digital Signal Processing or DSP is the true “four-year” commitment built upon all the powerful set of mathematical Fourier tools. What it basically does is: get an analog input (an analog signal); apply Fourier techniques to change it into a suitable digital form; change or tailor the digital version (“process” it) using a software program on a computer or a similar device, according to particular demands; and use Fourier tools once more to yield an analog output in the form of an electric signal capable of acting upon our everyday world.

Any student venturing into this DSP field will find out that there is even more math in this area than only traditional Fourier techniques: since digital signals sampled exclusively at fixed time intervals are to be dealt with, the so-called “discrete (non-continuous) Fourier transform” or maybe the “$z$ transform” will be needed. We will not go into any further detail about those additional mathematical tools, but we should only understand that they help analyze and synthesize digital systems by looking into such important details as stability and optimal component use.

6.1 Basic DSP Systems and Digital Filters

Figure 7 conveys this general idea: since our everyday world is perceived in analog form, almost all information coming from it will originally be in analog form. We also need an analog signal in order to produce some action in this kind of world. But it is by far easier and more efficient to decide what action to take by using a software approach (a program) than by reshaping the input signal directly into the output by purely analog means. This is what DSP is all about: capture an analog signal; convert it to a digital one (by means of an analog-to-digital converter or ADC); use this converted signal as input to a program; get the digital result; and convert it back to an analog response (by means of a digital-to-analog converter or DAC) capable of producing the desired action upon the analog world. The “signal reshaping” we have referred to in the previous paragraph is called “filtering”. A DSP system could therefore be also called a “digital filter”.

The very first electric analog filters were used in the field of telecommunications (in telegraphy) since the end of the XIX century, so this can hardly be labelled “new technology”, even though the first digital-filtering realizations come from the 1970s!

6.2 DSP Versus Traditional Analog-Processing

Of course, before the widespread use of microprocessors during the 1980s, all commercially available electric or electronic filtering took place exclusively by way of analog means: the only filters available outside from research laboratories, capable of dealing with electric signals, were analog.

Presently, however, it is far more flexible and cost-effective to implement signal-reshaping or filtering processes by means of a digital filter. Also, the developer need only be concerned with all the variables in a specific problem from a programming point of view: after the transceiver and ADC have done their work, the signals are now programming variables and their original physical nature is absolutely irrelevant. Then again, after the program yields a result, the problem of returning this to the external analog world in a usable form will be assigned to the commercially available DAC rather than to the program or developer. Even if this output signal were to drive a high-power (or non-electrical!) application, it is rather easy to amplify and change it as needed once its weaker electric version has already been delivered by the DAC.

Only if the signal that one needs to process changes very rapidly (is a “high-frequency” signal), one will still needs an expensive and bulky analog filter: the processing time will cause too long a delay for the response to be useful. Fortunately, those high-frequency real-time applications are quite specialized, while the majority of useful signals can be comfortably processed by inexpensive digital filters running on almost any computer or microprocessor.

7. ARE WE INTELLIGENT YET?

The RFID example should be more than enough to show that there is actually no need to pass the Turing Test in order for an electronic application or device to be considered intelligent in a practical and extremely useful sense.

If any device is capable of storing a unique identification and giving it back upon request, in the form of an electric signal, it will be deemed to be intelligent for all practical purposes.
The working example that accompanies this paper is an extremely elementary, but hopefully clear realization of what has been said so far:

It uses physico-chemical properties to provide an “acidity” (pH) transducer based on two small coins made of different metals, and an almost trivial processing system to display in binary form, on four LEDs (Light Emitting Diodes), up to 16 different levels of “acidity”.

Even though the example is presently lacking a DAC to turn the digital answer displayed by the LEDs back into an analog variable capable of producing direct action on the external world, one can easily imagine that, by providing a suitable commercially available DAC, this elementary system could be used, for example, to drive a motor and close or open a water faucet in order to increase or decrease the acidity of the lemonade that the whole system has been designed and programmed to prepare.

8.1 The Hardware

Surprisingly enough, the hardware is basically a single inexpensive chip: a so-called PIC (“Programmable Integrated Circuit”) by Microchip Technology Inc. [2]

The rest of the electronic components are merely supporting hardware: the four LEDs with their corresponding resistors to avoid excessive currents in the output display, a voltage source (battery), a temporary supporting board (which could be very easily replaced by a permanent printed circuit board), and some connecting cables.

8.2 The Software

The software might be even more trivial: it is written into the PIC memory using a few lines of standard C language instructing the PIC how to initialize and use every one of its input/output pins; how to read the analog input and convert it into a digital variable; and how to output this variable to light the green LEDs to represent the relative magnitude of the input in binary form; or to light the red LED instead, whenever the probes are disconnected.

Just for the sake of clarity, figure 8 shows the simple hardware circuit, while figure 9 shows the core of this “program” listed as the so-called “Decide” subroutine.

9. CONCLUSIONS

In conclusion, persons trained in software development could easily obtain information from a hardware component, regardless of how bulky or old-technology, “hard”, or “dumb” it actually is, by keeping in mind that the only tricky part involved is the choice of a transducer. Once this is done, an electric signal becomes available, and an ADC could then easily integrate said signal into the realm of information processing by means of a digital device perfectly within the scope of what the software engineer or technical person has been trained to do.

As soon as the electric signal is converted into a digital format (remember: transducers and analog-to-digital converters are all over the marketplace!), the physical variable might no longer be regarded as such, but rather as a variable (or data) in a programming context, and we certainly know how to handle that!

The additional good news is that there are presently thousands of commercially available transducers for (almost) any imaginable physical phenomenon, and thousands of analog-to-digital (and digital-to-analog converters), so capturing the status of a given device, machine, or environment, and taking it into our own field of expertise could be more a matter of patience browsing through a catalog, rather than a matter of extensive research on a different and unfamiliar technology.

10. ACKNOWLEDGEMENTS

My thanks to Jalasoft for granting me access to research resources, including valuable office time. Thanks to TechZone for the opportunity of sharing those ideas.

11. REFERENCES


http://crl.ucsd.edu/~saygin/papers/MMTT.pdf

[6] Ibid. 62
[7] Ibid. 65


[10] Ibid. 64