



SURVEY ON A RECONFIGURABLE SMART SENSOR INTERFACE

Krushnali A. Taware¹, Dr. Nita A.Doshi²

^{1,2}*Electronics & Telecommunication Engineering, SVPM's COE, Malegaon(Bk), India*

ABSTRACT

In wireless sensor network (WSN) a sensor interface device is very important. In the Internet of Thing(IoT) environment, every sensor is associated with the device is required to write a complex & long data collection program code. This paper deals with smart sensor interface: a portable STIM block is obtained using a VHDL model of IEEE1451.2. Hence compact solutions allowing low-cost, small, smart, large- scale production is possible. In order to evaluate the performances of proposed model, the comparison between microcontroller & VHDL-based STIM is done.

Keywords: *IoT, STIM, VHDL, IEEE1451.*

I. INTRODUCTION

In WSN involves a lot of different types of sensors. The different users have their own different applications and that requires different types of sensors. The interface devices are very specialized in working style. Meanwhile, the universal data acquisition interfaces are often restricted in physical properties of sensors like the signal type, connect number and sampling rate. We can use the micro control unit (MCU) is utilize as the core controller in data acquisition interface device. MCU has two advantages 1) low power consumption 2) low price. But, it performs a task by way of interrupt, which makes multi sensor acquisition interfaces not really parallel in collection of multisensor data. On the other hand, CPLD has unique hardware logic control, real-time performance, and synchronicity [2] which empower it on attain parallel securing for multi sensor information Furthermore extraordinarily move forward ongoing execution of the framework. In WSN, different users have their own applications that require different types of sensors and different sensor has its own requirements for readout and IoT (Internet of Things) may be another making of the web.

In WSN application sensor data acquisition surface device is the key part of study. The IEEE Electronic Engineering Association has also launched IEEE1451 smart transducer (STIM) interface standard protocol suite for the future development of sensors in order to standardize a wide range of intelligent sensor interfaces in the market and solve the compatibility problem of intelligent sensor [2]. The STIM interface standard IEEE1451 enables sensors to automatically search network. But, the sensors with the protocol standard have a high cost and still lack popularity in industrial WSN in IoT environment [2].This paper portrays an outline of a reconfigurable advanced mobile sensor interface for WSN for IoT nature's domain. This plan displays large portions points of interest as portrayed beneath.

1) CPLD is utilized Likewise the center controller acknowledge positively parallel procurement about sensor information Also arrival those confinement on the widespread information securing interface.

2) The design system applies the IEEE1451 interface protocol standard that is used for smart sensors of automatically discovering network [2]. Those capacity of plug Furthermore assume is utilized for the sensors not In view of IEEE1415 protocol standard, and the information procurement interface framework might accomplish

In this paper, this design take full advantage of CPLD characteristics. The design adopts IEEE1451[1],[2] smart transducer (STIM) interface standards, is adapted in this paper. Some vendors supply smart sensors adherent to IEEE1451.2, the standard part that describes Smart Transducer Interface Module (STIM), Transducer Electronic Data Sheet (TEDS), and Transducer Independent Interface (TII) [2]. Generally, these sensors have a microprocessor-centered architecture, where the CPU is devoted both to handle sensing element signal and to support IEEE1451.2 structure. But, it requires to write a complex code.

Fig 1 indicates the CPLD equipment square outline. We outline an reconfigurable smart sensor interface device that integrates information gathering utilizing simple Also advanced sensors, information transforming Toward utilizing CPLD as center controller, Also wired alternately remote transmission together. The device can be widely used in many application areas of the IoT and WSN to collect various kinds of sensor data in real time[2]. In CPLD we use a of system ip center module about IEEE1451. 2 relating protocol. Therefore, our interface device can automatically discover sensors connected to it, and to collect multiple sets of sensor data intelligently, and parallel with using high-speed RAM. CPLD performs control data acquisition, processing, and transmission intelligently, and make some preprocessing work for the collected data. Inside the CPLD the driver of chips on the interface device is also programmed.

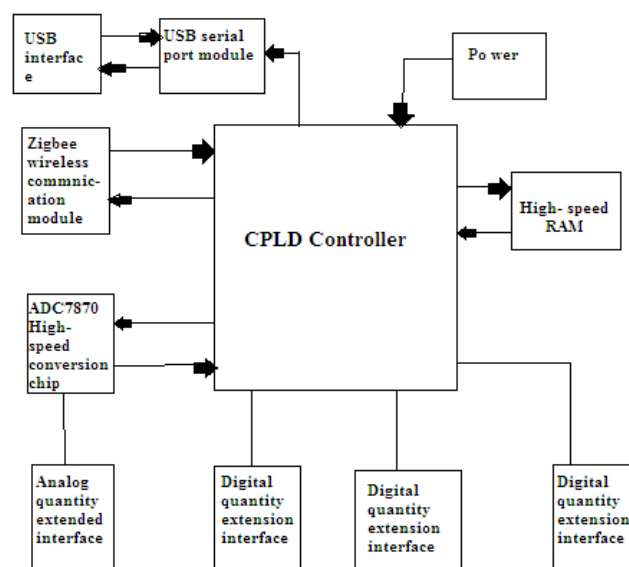


Fig.1 CPLD hardware block diagram.

Multiple scalable interfaces can be extended to 8-channel analog signal interface and 24-channel digital signal interface. This ensures that our device can connect with a number of sensors among the application of IoT or WSN and guarantees the diverse collection of the information[2]. Our design can achieve data transmission

through wired communication via Universal Serial Bus (USB) interface and wireless communication through Zigbee module. Therefore, we can choose different transmission mode. Analog to digital converter(ADC) is used to convert analog signals from analog sensors into digital signals to be processed by CPLD core controller.

II.THE PROPOSED APPROACH

The STMI has a modular structure. An analysis of the IEEE1451.2 indicates how those STIM collects information starting with sensor, holds sensor provision Furthermore applies alignment laws. The communication part of Network Capable Application Processor (NCAP) and the smart transmitter STIM is the sensor independent interface TII. Fig 1 is the TII function diagram, TII is easy in concept. Because of the interface with handshake lines DOUT (Data Out), DIN(Data In), DCLK (Data Clock) NCAP is basically a synchronous serial interface, as shown in Fig. 1. As slave device system, STIM remains in the standby state, until awaking notification is received from master device [1]Those STMI could perused alternately compose information through NIOE transport looking into TII alternately might initiate particular sensor through NIOE line indicator on TII. If NACK line STIM makes reaction data. Occasions might additionally make signaled with NCAP toward method for a intrude (NINT) [2].

The conceptual block diagram of proposed VHDL STIM is shown in Fig.3. The functions of intelligent transducer is mainly controlled by three fundamental state machine modules, which are Main or Master State Machine, Data Transport State Machine, Channel Trigger State Machine and in addition, memory controller with high speed RAM data cache is also involved. The Master State Machine controls effect on Data Transport State Machine and Channel Trigger State Machine and also is responsible for all the primary operations of the STIM .The Data Transport State Machine is responsible for data communication between the upper levels. Through the TII Data Transport State machine can transmit bit, byte and frame data.

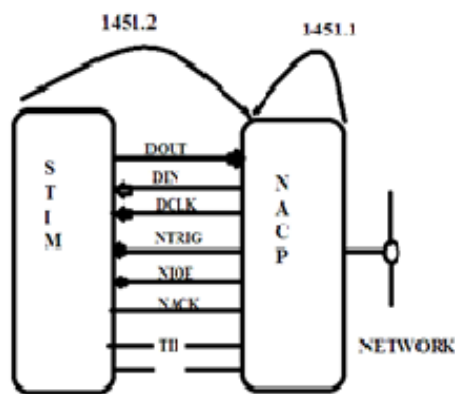


Fig 2. General model of IEEE1451

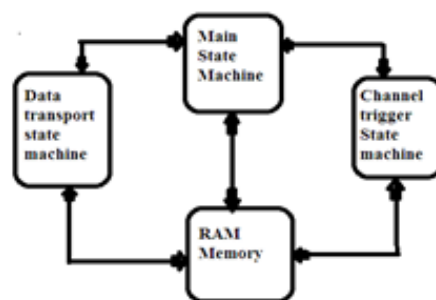


Fig 3. Block diagram of proposed STIM

Fig. 4(a), the VHDL STIM model describes only the block that has been called “Stand alone STIM.” According to IEEE1451.2, no hypothesis is done about how transducers and signal conditioning work. The proposed software model represents a transducer as a “digital number” contained in a register. How this number reflects detected physical amount (e.g., its change figure, nonlinearity, and so on.) is not concerning the present work.

The STIM model, as described here, is not supposed to introduce degradation of metrological characteristics of transducers since it does not apply any kind of mathematical transformation on transducer values. In order to demonstrate the feasibility of the proposed VHDL model, a complete prototype has been built. In addition, two programming based STIM models, acknowledged on a microcontroller, have additionally been considered to make an examination: a business execution proposed by Microchip in the AN214 application note and an enhanced rendition we understood.

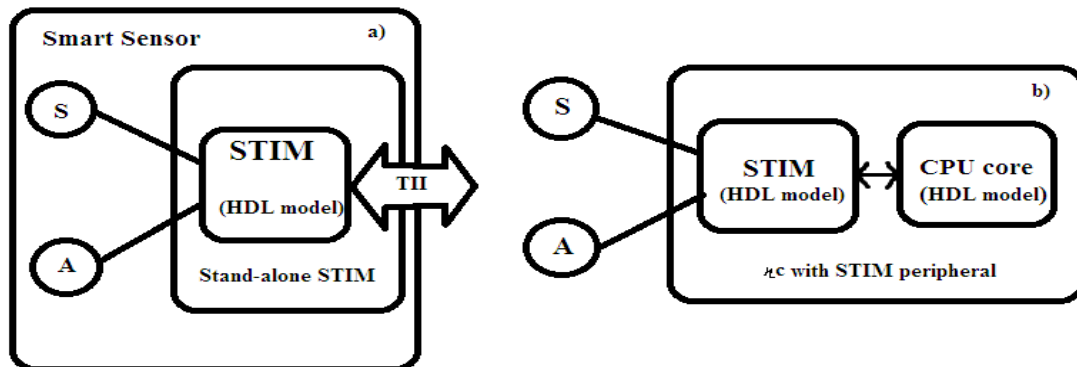


Fig.4 The STIM HDL model can be both (a) standalone (b) included into a microcontroller Sensor S and actuator A have been attached to STIM.

2.1 VHDL Configuration

Very-High-Speed Integrated Circuit Hardware Description Language (VHDL) design of the system consists two parts.

1) First should compose related characteristics of the reconfigurable advanced mobile sensor interface gadget toward alluding of the standard from claiming IEEE1451. 2 understanding by utilizing those VHDL dialect. It reflects the distinction between reconfigurable advanced mobile sensor interface gadget Furthermore general information securing card, which need an incredible impact clinched alongside intellectual elite gathering sensor data [2].

2) Second is programming the interface driver based on VHDL hardware description language. It mainly covers programming of each hardware chip driver and sensor driver on the device. The recommended VHDL STIM obliges something like 850 microcells (55%) and 512 bytes (2%) for RAM, including counters Also rationale for recurrence estimations for channel 1, as described above. Electronic datasheets (TEDS) are maintained in a RAM bank that is initialized after each reset; initial values can be stored either in an internal EEPROM area or in an external serial EEPROM [2].

2.2 IEEE 1451

IEEE 1451.4 defines a physical connection (Mixed-Mode Interface, or MMI) that is alternately used for TEDS data and analog signals, on either 2, 3 or 4 wires. This adapts the Standard for use with a wide variety of sensors and actuators. Contained in the standard is a definition for a transducer electronic data sheet (TEDS), which allows the use of very small memories, through the use of templates. The small physical size of the memory device allows TEDS to be included in tiny, lightweight sensors. However, the low bit-count available in physically small memory devices dictates that only essential data be stored in an array governed by What is

1451.4, what are its uses and how does it work? The template defines the significance and units associated with the stored data and the mapping of the data in memory. Prior to storing data and upon reading back the stored TEDS data, the template guides, respectively, packing and unpacking the data. New templates may be written in the template description language (TDL) defined by the standard4.

III. EXPERIMENTAL RESULTS

The experimental setup is shown in Fig. 5. to compare the VHDL prototype with the PIC18F452 implementation a test bench has been assembled. By switching a selector the gas and vapor detection system can be alternatively connected to the STIM under test. On the other side, a C-based NCAP simulator, driven by a PC through a RS232 link, has been employed to issue commands to the STIM under test[1]. Timing characteristics of prototypes are very important, since these values are a fundamental part of the TEDS. In the following, timing measurements have been achieved by means of a logic state analyzer 1692A by Agilent, with a resolution of plus or minus 2.5 ns.

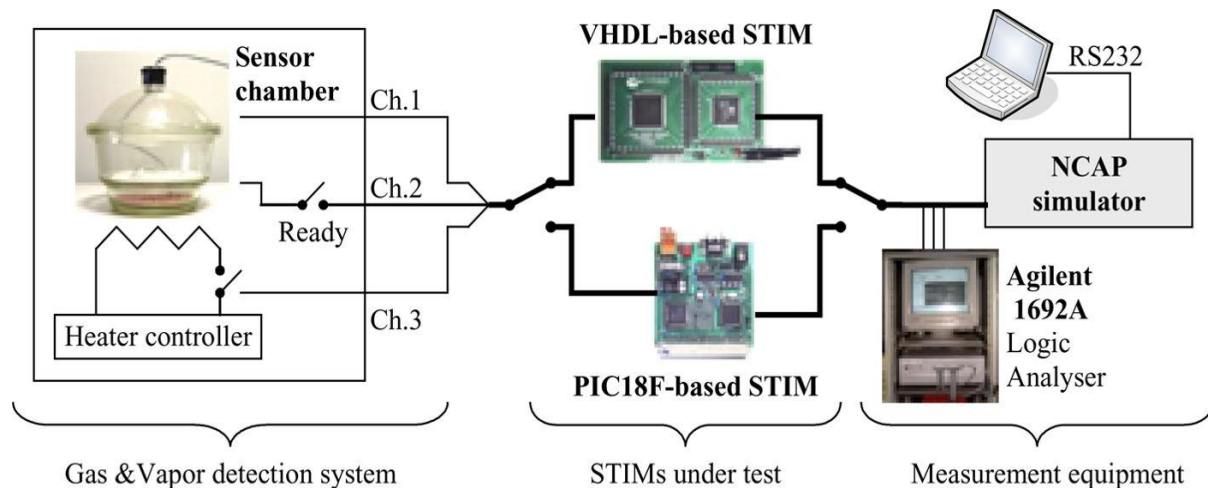


Fig. 5. Experimental setup. A vapor detection system has been connected to STIM by means of three channels. Selectors are used to choose the STIM to test. APC-driven NCAP simulator has been employed to verify TII connectivity, while a logic analyzer records time delays.[2]

Experimental tests have been conducted on TII link with the same communication clock DCLK (1.25 MHz) for all tested systems.

Table I reports average results, Standard deviation of time values, reported in Table I for the VHDL-based STIM, is always less than 10 ns; this is in accordance with internal STIM signal synchronization that operates at 40 MHz[1]. On other hand, timing of the microcontroller implementation has a standard deviation of 1µ s due to software polling variability. Data listed in column “AN214” have been derived from [12] as a further and useful term of comparison, although their standard deviation is not available.

Results clearly show that VHDL implementation is faster than the microcontroller ones, especially in the data transport state, where hold times greatly influence overall performances. These results depends on performances of the adopted CPLD; because of its “portable” nature, the model could be utilized even in different architectures, like other CPLDs or field programmable gate array (FPGA) or Application Specific Integrated

Circuit (ASIC). In the FPGA case, results may vary; probably the model requires fewer resources, thanks to the flexible FPGA architecture, but it could have slightly lower performance[1].

TABLE I

Parameter	VHDL-based STIM	PIC18F-based STIM	AN214 STIM
STIM Initial Communication Timing	75ns(max)	4.9µs	n.a
STIM Handshake Time(Data transport)	60ns	1.3µs	n.a
STIM Handshake Time(Trigger)	610ns	1µs	10µs
End-Of-frame Detection Latency	2.5µs	<10µs	50µs
Operational Hold-off Time	90ns	2.4µs	50µs
TEDS hold-off Time	90ns	50µs	1.2ms
Channel Update Time	2.7µs	<9µs	6µs
Channel write Setup time	0	5µs	16µs
Channel Read Setup Time	750ns	15µs	160µs
Channel Sampling Period	3.3µs	<10µs	160µs

TABLE I VHDL-BASED STIM IS COMPARED WITH TWO MICROPROCESSOR-BASED STIMS. LAST COLUMN REFERS TO MICROCHIP APPLICATION NOTE [12]

IV. CONCLUSION

This paper describes a reconfigurable smart sensor interface for WSN in IoT environment. The system designed based on IEEE1451 protocol by combining with CPLD and the application of wireless communication can collect sensor data intelligently. The application of CPLD makes the whole system more flexible and extensible and mainly simplifies the design of peripheral circuit. This paper shows a comparison between microcontroller and STIM, that with the advantage to be flexible a microcontroller-based STIM is a simpler solution since it is an easily software-programmable device. However, as shown in the real case we considered for the comparison, some extra logic could be needed to implement particular functions using microcontroller. The proposed VHDL STIM can achieve superior performances in a single-chip solution but it is objectively more complex, but it also occupies less silicon space. A VHDL STIM is suitable for mass-production of low-cost IEEE1451.2 compliant sensors. Also, a VHDL implementation could be convenient, compared with a microcontroller based STIM, if high performances are required or if a sufficient production volume is reached. Considering the fast evolving situation in the field of intelligent sensor, a reusable VHDL model can be quickly completed with any desired interface.



V. ACKNOWLEDGEMENTS

The author would like to thank anonymous reviewer for their constructive comments and valuable suggestions that help in the improvement of this paper. The author would also like to thank SVPM's College of Engineering for the facilities provided.

REFERENCES

- [1] A. Depari, P. Ferrari, Member, IEEE, A. Flammini, Member, IEEE, D. Marioli, and A. Taroni, "A VHDL model of a IEEE1451.2 Smart Sensor: Characterization and Applications", IEEE SENSORS JOURNAL
- [2] Qingping Chi, Hairong Yan, Chuan Zhang, Zhibo Pang, and Li Da Xu, Senior Member, IEEE) Formats, IEEE Standard, "A Reconfigurable Smart Sensor Interface for Industrial WSN in IoT Environment".IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, VOL. 10, NO. 2, MAY 2014
- [3] IEEE Standard for a Smart Transducer Interface for Sensors and Actuators—Network Capable Application Processor (NCAP) Information Model, IEEE Standard 1451.1-1999, 1999.
- [4] M. Svěda and R. Vrba, "An integrated framework for internet-based applications of smart sensors," IEEE Sensors J., vol. 3, no. 5, pp. 579–586, 2003.
- [5] IEEE Standard for a Smart Transducer Interface for Sensors and Actuators— Mixed-mode Communication Protocols and Transducer Electronic Data Sheet (TEDS) Formats, IEEE Standard 1451.4, 2004.
- [6] S. Li, L. Xu, X. Wang, and J. Wang, "Integration of hybrid wireless networks in cloud services oriented enterprise information systems," Enterp. Inf. Syst., vol. 6, no. 2, pp. 165–187, 2012.
- [7] Q. Li, Z. Wang, W. Li, J. Li, C. Wang, and R. Du, "Applications integration in a hybrid cloud computing environment: Modelling and platform," Enterp. Inf. Syst., vol. 7, no. 3, pp. 237–271, 2013.
- [8] L. Wang, L. D. Xu, Z. Bi, and Y. Xu, "Data cleaning for RFID and WSN integration," IEEE Trans. Ind. Informat., vol. 10, no. 1, pp. 408–418, Feb. 2014.
- [9] Y. Fan, Y. Yin, L. Xu, Y. Zeng, and F. Wu, "IoT based smart rehabilitation system," IEEE Trans. Ind. Informat., vol. 10, no. 2, pp. 1568–1577, 2014.
- [10] W. He, G. Yan, and L. Xu, "Developing vehicular data cloud services in the IoT environment," IEEE Trans. Ind. Informat., vol. 10, no. 2, pp. 1587–1595, 2014.
- [11] M. T. Lazarescu, "Design of a WSN platform for long-term environmental monitoring for IoT applications," IEEE J. Emerg. Sel. Topics Circuits Syst., vol. 3, no. 1, pp. 45–54, Mar. 2013.
- [12] R. Fisher and J. Burch, "The PICmicro MCU as an IEEE1451.2 Compatible Smart Transducer Interface Module," 2000, AN214, Microchip
- [13] R. M. Haralick, K. Shanmugam and I. Dinstein "Textural features for Image Classification", IEEE Transactions on Systems, Man and Cybernetics, Vol.3, pp. 610-621, November 1973