

Reconfigurable STIM For Robots In Iot Environment

Y. Bharath kumar¹ and S. Ilaiyaraja²

*¹Course of Embedded System Technologies, Velammal Institute of Technology,
Panchetti-601204, TN, India*

*²Associate professor, Velammal Institute of Technology, Panchetti-601204, TN, India
¹ybkreddy92@gmail.com, ²silaiya@gmail.com*

Abstract

Robots need a sensor interface for getting the environmental data. In this paper, the smart sensor interface with reconfigurability is used in IoT environment. The connect numbers, signal types and sampling rate can be increased. The Field Programmable Gate Array (FPGA) is taken as the core controller. By this, the data can be read parallelly and in real time. These also work with high speeds and collect multiple sensor data. It is smaller in size and has fine grain capability. It also works faster than CPLD. IEEE1451.2 protocol is implemented here in this design. The smart sensor interface hardware is simplified into a program which can be done in single reconfigurable chip. Software design and the concerned protocol is realized for acquisition of sensor information. This is a new trend that is followed instead of traditional sensor data acquisition. The sensors are combined with new FPGA programming technology with the IEEE1451.2 standards for intelligent acquisition. These robots work in IoT environment hence they can communicate each other collectively. It is practically applied in disaster management control with good performance and it can be also used in industrial and medical environment.

Keywords: FPGA, IEEE 1451.2, STIM, Internet of Things, Robot.

Introduction

FPGAs are reconfigurable devices that are been used in variety of applications and these can be implemented in IoT environment. FPGAs have millions of logic blocks and can be programmed by hardware description languages like Verilog HDL and VHDL. Sensor data acquisition is done in faster manner and in real time without any delays. It enables us to parallel process the sensor data from the robot. These devices are perfectly suited for the wireless sensor networks. Hence these have parallel processing capability and they can operate quicker way. In order to gather multiple

sensor data at a time, these type of devices can parallel process the data. FPGA has fine grain capability and can give high resolution signals. FPGAs can adapt for control embedded systems and also used for many complex applications. Every sensor and actuator have their own working principle thereby, they need drivers which are in cumbersome manner and are interfaced to the logic controller.

Wireless sensor networks are used majorly for gathering the sensor data and used in many applications such as surveillance, health care monitoring, medical monitoring, environment monitoring and also in military applications. WSN when used under IoT environment it can increase its range of applications. WSN enables to acquire the information from the sensors implemented in the robot. Types of sensor can be detected by sensor interface device and it can be programmed by HDL for FPGA. Thus, the surrounding statistics can be collected by sensors with particular sensor assigned. Multiple details can be collected at a time with WSN and parallel processing with FPGA. Finally, WSN can get diverse range of information from the sensor in IoT environment.

Intelligent sensors are interfaced with the logic controller and have compatible problem. It can be solved by using IEEE 1451 standard named STIM (smart transducer) interface standard protocol launched by IEEE Electronic Engineering Association. This protocol defines the sensor interface for the data acquisition. This protocol searches the network automatically for the sensors, STIM also contributes to improve the sensor network in the robot [14]. Such type of sensors with these standards are costly and are less popular. They are also limited in market, so an alternative dedicated hardware interface is been proposed by IEEE. Diverse sorts of sensors can be interfaced by this IEEE 1451 standard and can be created in IoT environment.

This paper has designed for sensor networks with re-configurability. The sensors in robots are parallel processed in real time and are employed under IoT environment. The application scope of the robot is expanded with actualizing WSN and IoT any sort of sensor can be supplanted whenever it can acknowledge any kind of sensor recorded. Any type of sensor can be replaced at any time, it can accept any type of sensor listed. This design has better compatibility. Robots can roam freely without any manual interruption also, these robots also can be controlled manually if needed. It gathers the sensor data and transmits to user location, which is in IoT environment it can be controlled and monitored anyplace wherever internet is available with respected user authentication.

Relating IOT With WSN In Robot

IoT is designed for our future, these interconnect objects based on standard communication protocols and these objects are addressable. Both machine to machine and person to computer can communicate with each other. Here in this paper the data from sensor networks can be interfaced to the internet, all of the devices are given their own addresses for communicating between them. Many sensor networks are related together to get huge data from the environment [13]. Collecting the data is an

important act in this paper and this is done by the sensors that are implemented in the robot.

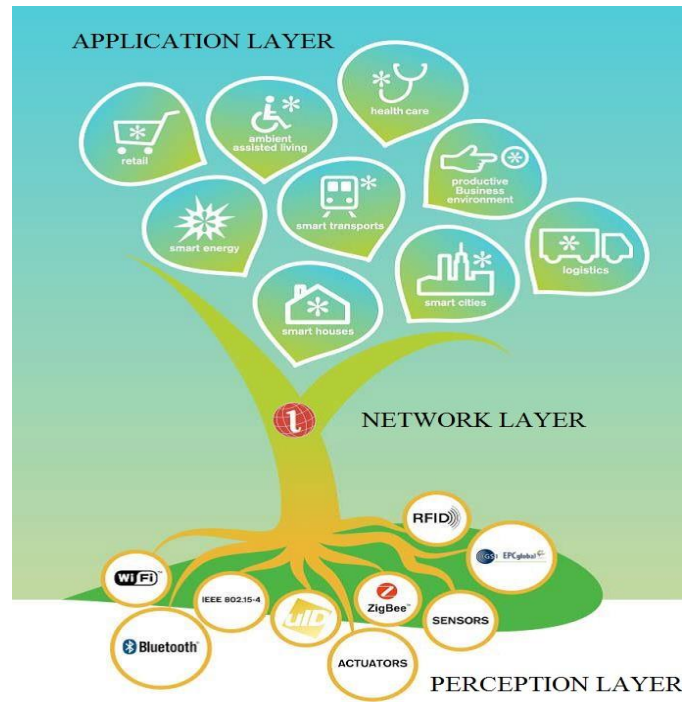


Figure 1: IOT Basic Structure

There are many wide varieties of applications for IoT in future such as structural health, smart parking, smart roads and waste management etc. Fig. 1 shows the IoT basic structure it has wide acceptable of modules in perception layer. Through network layer it can make these modules function in wide area of applications. IoT architecture has three layers application layer, network layer and perception layer [1]. The layer descriptions are as follows. The perception layer consists the end objects such as RFIDs, camera, sensor networks and data acquisition. Data acquisition gathers the data from the environment and networks it through WSN [10]. In this paper we discuss about this data acquisition. The network layer describes the type of network that we use it, may be broadband, 2G, 3G or any other network that is used [9]. In this layer the data is saved in clouds, it provides cloud services for the sensor networks. Application layer is the user interface by which the user monitors or controls the sensors and the actuators. It can be any application such as medical, environment, disasters, etc.

Normally robots have many number of sensors in them to get the data and get actuated, these sensors are interconnected with IEEE 1451 and such robots work under IoT environment to gather the huge data from many places at a time. These also work co-operatively for single application in different places efficiently [2].

Architecture

A robot having smart sensor interface with re-configurability does the data acquisition for the sensors in robot and actuates the motors for movement of the robot. This system can be used in wide range of applications in areas such as IoT and in WSN for collecting the sensor information in real time and in faster manner.

In FPGA using Verilog HDL IEEE 1451.2 protocol (STIM) is programmed for interface [4]. The sensors are automatically discovered by the FPGA hence STIM protocol has been coded in FPGA. The sensors collect multiple data at a time intelligently and parallel process it to achieve higher speeds.

FPGA is the logic controller programmed for interfacing the device. It is used to process the data intelligently, acquire the data, process and transmit the data. It also preprocess the signals after collecting the data [3]. The controller is designed for getting multiple scales from the sensor as an equipment. The number of channels are extended by adding ADC chip. These can have 8 channel analog interface, 24 channels for digital interface. We can make sure that wide collection of data can be done by connecting with number of sensors in our device.

Here zigbee wireless module is used for data transmission from the sensors to the user module and communicating between robots and the control unit. By wired we can have it by using USB or with RS232.

In this design the controller collects the analog data from the sensors like color, humidity, PIR sensor, proximity sensor etc. Digital sensor data are also collected and transmitted e.g., temperature sensor. The FPGA controls the ADC module, it also interfaces with FPGA by that it can collect 8 channel and 24 channel for analog and digital signals respectively. The data is processed by SRAM to process the data faster. The data is then transmitted to the remote module by USB and with zigbee wireless communication also [7]. The robot can be controlled by the remote control module handled by the user. This is also processed by the FPGA controller. The design has camera which is interfaced with FPGA and this video can be viewed remotely by the host computer that is, by remote control [12]. The video can also be stored for future purpose.

These robots work in IoT environment and they interact with each other and pass there information to the IoT management Centre and the information is passed on to the required destination as per application implemented.

Implementation

A. Hardware Architecture

The hardware architecture has two modules remote robot and host computer user. Remote robot hardware comprises the main logic controller FPGA (EP3C16F484), peripheral devices and supporting hardware. For power supply 7.5V adapter is used. The FPGA is interfaced SDRAM with SDRAM controller, it has 8MB memory. The sensors are interfaced with ADC (ADS7870) chip having 8 channels with high speeds. The extended interface can expand the quantity of sensors utilized. The robot

FPGA is also interfaced with DC motor control with PWM output with FPGA for movement of the robot [11].

Fig. 2 shows the hardware block diagram. The robot is able to send and receive the information via zigbee wireless communication. It sends the sensor data to the host computer, the host computer sends the control signals to the robot. The robot also takes its own decision from the sensor data. Zigbee module is interfaced with GPIO or with mini-USB interface.

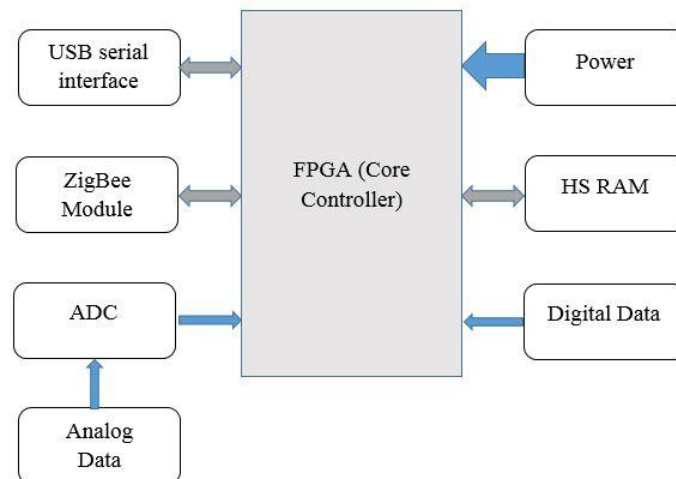


Figure 2: Hardware Architecture

The sensor data is also preprocessed here in the robot and send to the host computer. The host computer has a VGA interface, LCD display and control switches, it also has the connection with the IoT environment. It communicates with the robot with zigbee wireless communication module.

B. Verilog HDL Design

The proposed Verilog STIM model is implemented by using DE0 Nano development board. These have large number of RAM blocks gives flexible architecture and reduces the complexity. Fig. 5 shows the states and there flow is described below.

- State A: Waiting state - Waiting for the trigger.
- State B: Trigger ON – Filtered input is collected.
- State C: Memory Request – Awaiting request for accessing memory.
- State D: Memory Update – Data is given as input to the memory.
- State E: Memory Release – Releasing memory and signaled as data valid.
- State F: Trigger OFF – Trigger is switched off.

Verilog Hardware Description Language codes is split into two modules. One has the basic re-configurability features for smart sensor interface with reference IEEE 1451.2 standard. It brings the difference between the regular data acquisition and the smart sensor interface with re-configurability. The other module collects the sensor drivers and the actuator driver for the sensors and actuators in the robot [8]. It also

has the interfacing codes for its peripherals such as VGA, PS2 keyboard, LCD, SDRAM interface etc.

IEEE 1451.2 protocol is programmed using Verilog HDL language to collect sensor information and having sensor compatibility with IEEE 1451.2 standards [5], finally forming the smart sensor interface with re-configurability. This program can accept different kind of available sensors. Hence, intelligent sensors are not needed by this, the cost of the implementation can be reduced.

Fig. 3 shows the structural model of Verilog STIM. Here STIM is designed by the Verilog code for overall design to realize the intelligent transmitters. It contains the spreadsheet TEDS (Transducer Electronic Data Sheet) [6], channel setting module, TII (Transducer Independent Interface). Intelligent transducer can be realized is mostly by supervising the state machine modules. The state machines are control state machine, data communication state machine, and channel setting state machine. Scheduling is done by the control state machine and other two state machines is controlled by the control state machine.

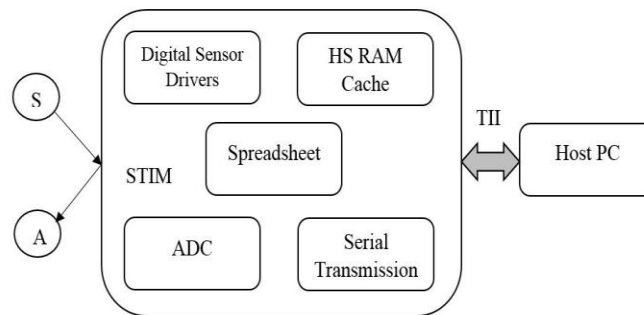


Figure 3: Verilog HDL Structure Diagram

The control state machine switches the process every state in STIM, it manages both data communication and channel setting. As shown in fig. 4 the control state machine also interfaces with high speed RAM data cache to increase the reliability of the transmission. The data communication state machine gets contribution from TII to transmit the data after processing the sensor values. The top layer user application software controls the state machine by sending signals. Encoding and decoding of memory address of each sensor is done by the state machines to operate different types of sensors.

Channel setting state machine module gets the data from different sensors in various channels. The channel setting state machine are independent with each other. These are hidden from the sensor channels.

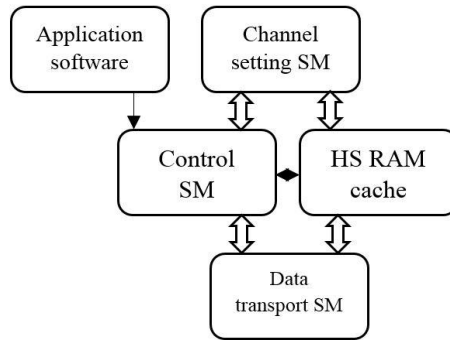


Figure 4: STIM Structure For State Machines

Channel setting machines are started with its related setting signal or it will be resting. All sensors channel setting state machine have identically same build. Parallel process is executed with having less delay. Data collection have higher reliability giving real time performance. It processes the multiple signals at a time acquired by different sensors. It stores the recently processed transmission data in the cache for getting higher speeds.

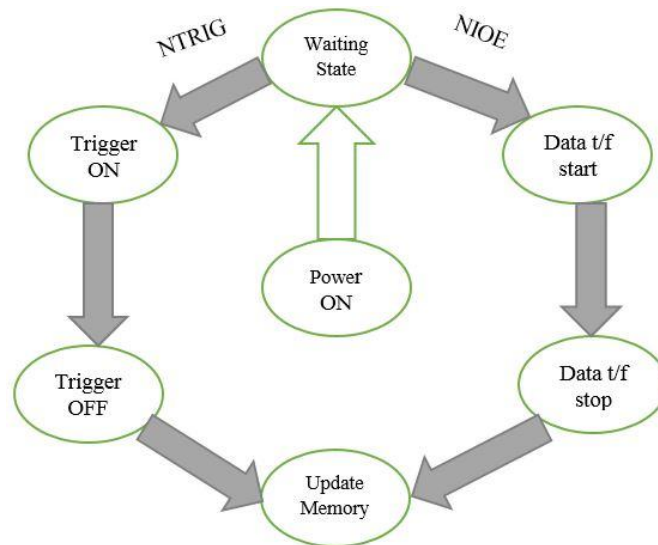


Figure 5: STIM State transition

TEDS module is designed separately with independent circuit for IP in FPGA. It consists the details for different types of sensors. It gets the data and removes the improper or wrong data that is been collected by the sensor. It compares the data gathered with the corresponding sensor type. If the details are matched then they are transmitted, otherwise it sets the channel again for getting the next data. Data length can be different according to the priority. Conversion rate depends on the data length.

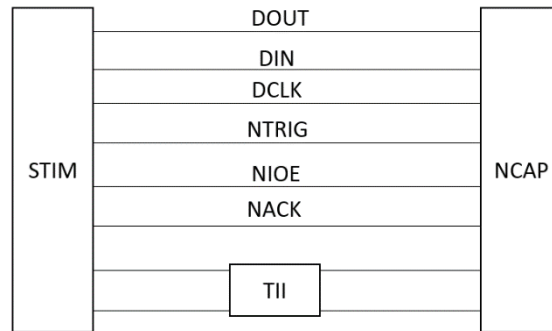


Figure 6: Network Interface With Independent Sensors

Transducer independent interface can be able to connect with internet using NCAP (Network Capable Application Processor). Both the STIM and NCAP are connected with each other, for handshaking. The data is exchanged with each other. The handshake lines are shown in fig. 6. With the usage of clock synchronicity is achieved in by the design. TII gets contribution from RS232. It works with master and slave principle. Slave waits in standby state till it gets acknowledgement from the master. STIM acts as the master device in this system and controls the communication done by NCAP. Control signals and feedback signals are done with DIN and DOUT cables. The final proceeding is done by user application software. Driver programs are coded in FPGA for interfacing the ADC chip, VGA, and other parameters needed for getting the data and control signals to and fro from it. Fig. 7 shows the interface between the ADC and the FPGA, by this interface the FPGA can collect the data from analog sensors also.

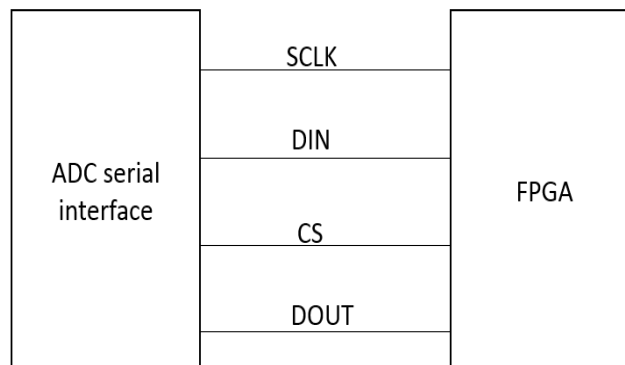


Figure 7: ADC interface with FPGA.

Results and Comparison

As per block diagram shown in Fig. 2. The Verilog prototype is compared with PIC18F execution. The humidity, temperature and pir sensors are connected alternately to STIM, also by switch they get selected. The microcontroller based NCAP is simulated by RS232 and results are obtained. Table 1 gives the results and

shows the comparison between the PIC and the Verilog model. From this table, we are able to know that Verilog module is faster than the traditional microcontrollers.

These FPGA model also reduces the power and size of the whole implementation setup. Example snapshot of data transport is shown in Fig.8. It shows communication link between Verilog STIM and NCAP. Here RAM cache is used for TEDS this give higher speeds of transport and has good performance. The PIC18F also has good performance due to its software architecture. Based on the type of FPGA or CPLD used the performance and speed differs.

Table 1: Comparison between Verilog and PIC based STIM

Parameter	Verilog Based STIM (FPGA)	PIC18F Based STIM
STIM Initial Communication Timing	70 ns	4.9 μ s
STIM Handshake Time (Data Transport)	65 ns	1.3 μ s
STIM Handshake Time (Trigger)	680 ns	1 μ s
Channel Update Time	2.9 μ s	<9 μ s

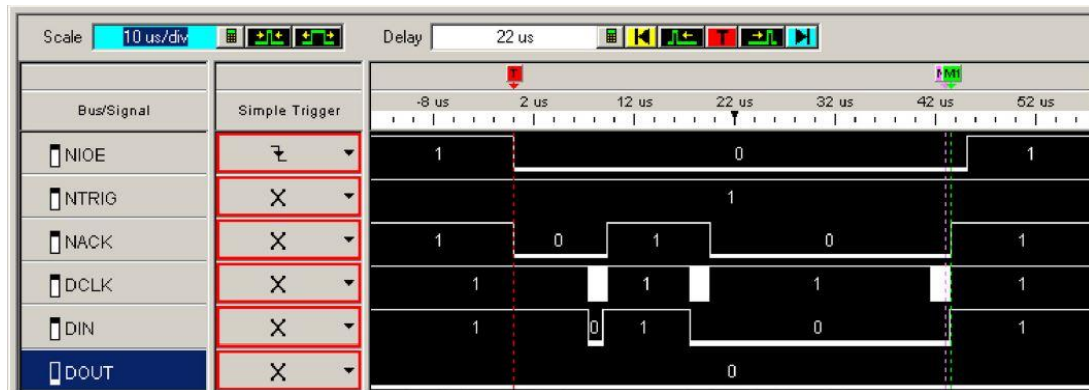


Figure 8: Data transport snapshot (example)

Conclusion

This paper comprises the wireless sensor network in robot having smart sensor interface in them. They work in IoT environment, they collect the information from sensor intelligently hence it uses IEEE 1451.2 protocol. The STIM is applied in FPGA with wireless communication. The sensors in robot are parallel processed and work in real time. The sensor data acquisition is done with high speeds in IoT environment. The circuit is simplified by using FPGA gives re-configurability and it is flexible, also can be extendable. The sensor data is collected efficiently in

intelligent manner and makes a reliable transmission between the robot and host computer. Sensors are used until they are connected, these also have plug and play ability for the robot. So that application of the robot can be changed according to the situation. This robots can be mainly used for disaster affected areas. By expanding the spreadsheet we can increase the range of the applications. In future it can have more advancements such as robots can directly communicate through internet and monitored by the user also.

References

- [1] 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.
- [2] W.-K. Chen, *Linear Networks and Systems* (Book style). Belmont, CA: Wadsworth, 1993, pp. 123–135.
- [3] L. Benini, "Designing next-generation smart sensor hubs for the Internet of Things," in *Proc. 5th IEEE Int. Workshop Adv. Sensors Interfaces (IWASI)*, 2013, p. 113.
- [4] Y. Chen and V. Dinavahi, "Multi-FPGA digital hardware design for detailed large-scale real-time electromagnetic transient simulation of power systems," *IET Gener. Transmiss. Distrib.*, vol. 7, no. 5, pp. 451–463, 2013.
- [5] Smart Sensor Interface for IEEE 1451 [Online]. Available: <http://www.jlminnovation.de/products/ieee1451>, accessed on Aug. 16, 2010.
- [6] L. Bissi et al., "Environmental monitoring system compliant with the IEEE 1451 standard and featuring a simplified transducer interface," *Sens. Actuators A, Phys.*, vol. 137, no. 1, pp. 175–184, 2007.
- [7] W. Li et al., "The spreadsheet (TEDS) research in the IEEE1451 agreement," *Comput. Knowl. Technol.*, vol. 3–20, pp. 334–336, 2008.
- [8] P. Cheong et al., "A ZigBee-based wireless sensor network node for ultraviolet detection of flame," *IEEE Trans. Ind. Electron.*, vol. 58, no. 11, pp. 5271–5277, Nov. 2011.
- [9] J. H.-L. Lu et al., "A low-power, wide-dynamic-range semi-digital universal sensor readout circuit using pulse width modulation," *IEEE Sensors J.*, vol. 11, no. 5, pp. 1134–1144, May 2011.
- [10] O. Vermesan, P. Friess, P. Guillemin, S. Gusmeroli, H. Sundmaeker, and A. Bassi. Et al. Eds., "Internet of things strategic research roadmap," in *Internet of Things: Global Technological and Societal Trends*. Aalborg, Denmark: River Publisher, 2011, ch. 2, p. 52.
- [11] 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.

- [11] A. Myaing and V. Dinavahi, "FPGA-based real-time emulation of power electronic systems with detailed representation of device characteristics," *IEEE Trans. Ind. Electron.*, vol. 58, no. 1, pp. 358–368, Jan. 2011.
- [12] E. J. Bueno et al., "A DSP-and FPGA-based industrial control with high speed communication interfaces for grid converters applied to distributed power generation systems," *IEEE Trans. Ind. Electron.*, vol. 56, no. 3, pp. 654–669, Mar. 2009.
- [13] L. Li et al., "The applications of WIFI-based wireless sensor network I n internet of things and smart grid," in *Proc. 6th IEEE Conf. Ind. Electron. Appl. (ICIEA)*, 2011, pp. 789–793.
- [14] K. C. Lee et al., "IEEE-1451-based smart module for in-vehicle networking systems of intelligent vehicles," *IEEE Trans. Ind. Electron.*, vol. 51, no. 6, pp. 1150–1158, Dec. 2004.

27552

Y. Bharath kumar