International Journal of Engineering Research And Advanced Technology (IJERAT)
ISSN: 24546135 [Volume. 02 Issue.04, April-2016]

Data Procurement System Using Controller Area Network In Automobile Industry

Mr. S. Aravindh[1], K Suresh[2]
Associate Professor,
[1]Department of Computer Science & Engineering,
[2]Dept. Of Electronics & Communication Engineering
Gojan School of Business & Technology,
Red Hills, Chennai - 52

***ABSTRACT*:**

In order to reduce point to point wiring complexity in automobiles, Controller Area Network is suggested as a means for data communication within the subsystems in vehicles. The most important factor of CAN bus is that nodes can be added or removed at any time so that it offers increased flexibility and expandability. This paper describes the ARM7 based design and implementation of CAN communication for data acquisition system. It focuses on hardware design of CAN node. Hardware interface mainly consists of LPC2148 microcontroller from ARM 7 family, MCP2551 stand-alone CAN controller and MCP2515 CAN transceiver. The communication is implemented via an industry standard Serial Peripheral Interface (SPI) with data rates up to 5 Mb/s. Temperature sensor, Pressure sensor and Fuel sensor are used as the data collection agents and the collected data is displayed in a Personal computer for database management.

***Index Terms*:** *Controller Area Network (CAN), Fuel sensor, Serial Peripheral Interface (SPI), Pressure sensor.*

I. INTRODUCTION

In present scenario automation is needed in many systems for better performance. A modern vehicle may have several **Electronic Control Units (ECU)** for different subsystem. Some of the subsystems are cruise control, airbags, measurement of temperature, fuel level indicator and so on. There is a need for communication among these subsystems which can be provided by means of CAN protocol. The message is broadcasted to each and every node in the network at a transmission rate of 1Mbps. As it provides reliable communication between devices, it finds its application in automobile industry, medical field, industrial and factory automation and avionics.

II. CONTROLLER AREA NETWORK

CAN is synchronous, serial, multi-master bus developed to establish communication among embedded system. It uses carrier sense multiple access protocol which is very effective in transferring correct and authenticate data through the network. Since it incorporates bitwise arbitration technique, there is no requirement for using extensive collision detection circuits and random timers. **Open System Interconnection model (OSI)** was developed by International

Organisation for Standardisation to provide data transfer between various nodes. Out of the seven layers of the OSI model, the CAN protocol utilises lower two layers- **physical and data link layers**. Physical layer of the CAN and OSI model describes the actual communication between devices connected.

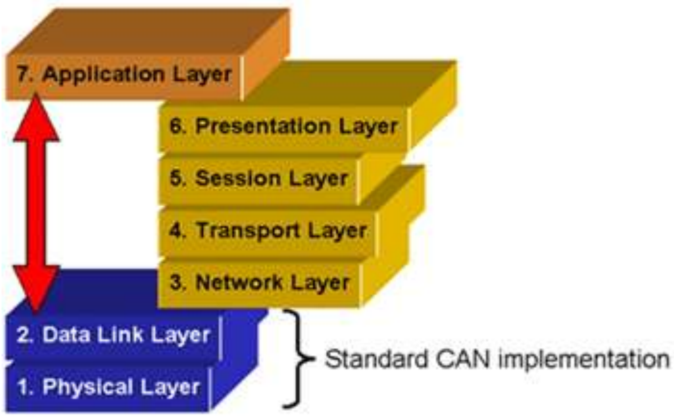

Fig: 1 ISO/OSI Reference Model

A.CAN- FRAME FORMAT

Standard CAN Frame format

S O F	11-bit Identifier	S R R	I D E	R T R	R 0	D L C	Data Upto 8-bit	C R C	A C K	E O F	I F S

Fig: 2 Standard CAN Frame format

The abbreviation of the fields of Fig: 2 are:

- SOF-It is single dominant start of frame (SOF) bit. Message in CAN starts with SOF.
- Identifier -The Standard CAN 11-bit identifier. It establishes the priority of the message.
- RTR–Remote Transmission Request (RTR).
- IDE-A dominant single identifier extension (IDE) bit. If this bit is enabled means that a standard CAN identifier with no extension is being transmitted.
- R0- Reserved bit
- DLC- The 4-bit Data Length Code (DLC).It has the number of bytes of data which needs to be transmitted.
- Data- Application data in the range of 64 bits may be transmitted.
- CRC- The 16 bit (15 bits plus delimiter) Cyclic Redundancy Check. It contains the checksum of message.
- ACK –Acknowledge bit
- EOF – End of frame (EOF) bit
- IFS- This 7 bit Interframe Space(IFS)

EXTENDED CAN FRAME FORMAT

S O F	11-bit Identifier	S R R	I D E	18-bit Identifier	R T R	R 1	R 0	D L C	Data Upto 8-bit	C R C	A C K	E O F	I F S

Fig: 3 Extended Frame format

- SRR–Substitute Remote Request (SRR) bit. The RTR bit in the standard message is replaced by it.
- IDE–A recessive bit in the identifier extension (IDE). It denotes that more identifier bits follow. It is followed by an 18-bit extension.
- r1–An additional reserve bit has been included ahead of the DLC bit following the RTR and r0 bits.

B. DATA COMMUNICATION IN CAN

Each node has both transmitter as well as receiver. This protocol allows each and every node in CAN to observe the bus network in CAN prior to transmitting a message. When there is no activity in the network, the sender of information transmits to all nodes on the bus. In addition to this, it allows collision to be solved using arbitration, based on a preprogrammed priority of each message in the identifier field of a message. This feature allows messages to remain intact after arbitration is completed even if collisions are detected. CAN define the logic bit '0' as the dominant bit whereas the logic bit '1' as the recessive bit. As “0” on the bus wins over “1” on the bus, losing node stops transmitting whereas winner node continues. All the nodes read the message and decide whether it is relevant to them or not. The transmission is acknowledged by the relevant node.

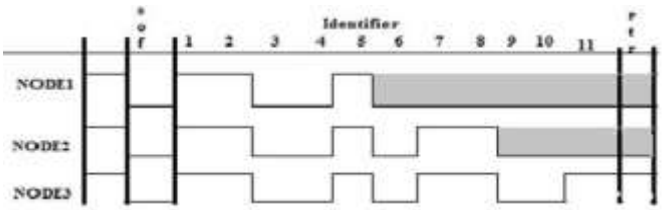

Fig: 4 Bus Arbitration

C. CAN NODE REQUIREMENTS

In CAN open there are unique addresses available for upto 127 nodes on the bus but practically it is limited to 110 nodes per bus. The CAN bus can have a maximum bus length of 1km with a bit rate of 10Kbps and that of minimum is 40m with 1Mbps. Each CAN node requires a host processor, CAN controller and CAN transceiver.

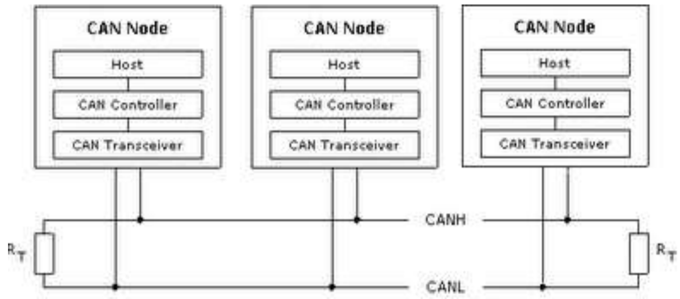

Fig: 5 CAN Bus with three CAN nodes

1. *CAN CONTROLLER (MCP2515)*

MCP2515 is an 18 bit pin package which controls every nodes on the CAN bus. It is a stand-alone CAN controller that implements the CAN specification, version 2.0B.It has the capability to transmit and receive both standard and extended data and remote frames. It stores the received bit serially until the entire message is available and it transmits it to the CAN controller. A one shot mode on the CAN bus ensures that the message transmission is attempted only once. An unpowered node or brown-out node will not disturb the CAN bus. In order to filter out unwanted messages MCP2515 has two acceptance masks and acceptance filters so that host MCUs overhead is being reduced. It is interfaced with microcontrollers via an industry standard Serial Peripheral Interface (SPI).

2. *CAN TRANSCEIVER (MCP2551)*

It is high speed, fault tolerant device that serves to provide interface between the Can controller and the physical bus. The MCP2551 imparts differential transmit and receive capability for the CAN protocol controller and is fully compatible with ISO -11898 standard, including 24V requirements. Each and every node in a CAN system must have a device to convert the signal generated by a CAN controller to signals suitable for transmission over the bus. MCP2551 CAN outputs will drive a minimum load of 45ohms.The Rs pin allows three modes of operation to be selected- High Speed, Slope Control and standby mode. High Speed mode is selected by connecting Rs pin to Vss. Slope Control mode further reduces EMI by limiting the rise and fall times of CAN_H and CAN_L. The device may be paced in standby or sleep mode by applying a high level to Rs.

3. *HOST PROCESSOR*

ARM LPC2148 is the most commonly used IC from ARM7 family. It is manufactured by Philips and is provided with many inbuilt peripherals making them more efficient and reliable for beginners as well as application developers. It is a 16/32 bit microcontroller with real time emulation and embedded trace support. It has a high speed flash memory ranging from 32KB to 512KB. Serial communication interfaces ranging from multiple UARTs, SSP, SPI, and USB Full speed 2.0 to I2C Bus to on-chip SRAM 8KB to 40 KB makes them well suited for communication gateways. Various PWM channels, single or dual 10-bit ADC, 32-bit Timers and 45 fast GPIO lines makes these microcontrollers suitable for industrial and medical applications.

III. BLOCK DIAGRAM

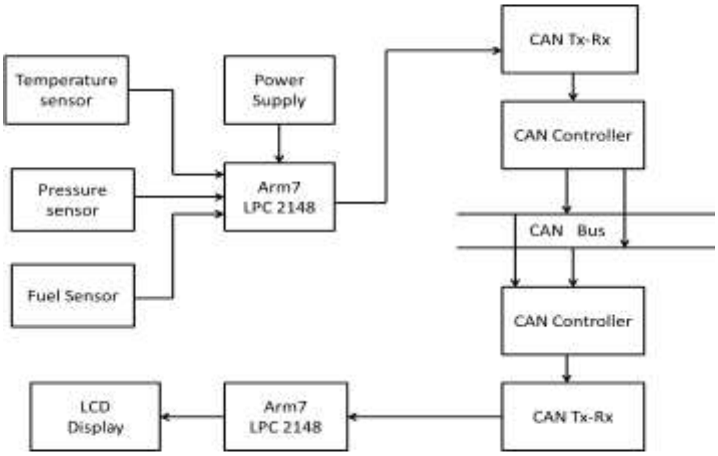

Fig: 6 Block Diagram of our proposed model

ARM as the host processor monitors the status with various sensors. The Temperature sensor, Pressure sensor and Fuel sensors and ARM processor acts as the data acquisition unit. The communication between these sensors is carried out by means of CAN bus. The sensed data will be passed to the microcontroller. The microcontroller will perform signal conditioning and ADC. From there the data will be send to CAN controller and then to CAN transceiver. In order to transfer the data to the monitoring unit, the CAN transceiver will load the data on to the bus and the collected data is maintained in a Personal Computer.

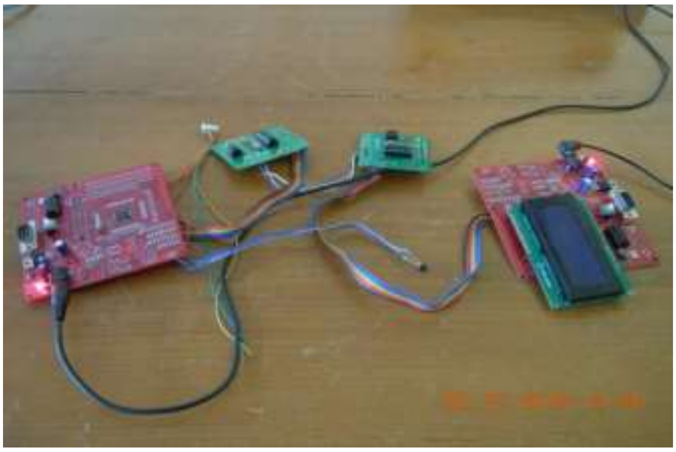

Fig: 7 Working Model

IV. HARDWARE COMPONENTS USED

- CAN Controller MCP2515
- CAN Transceiver MCP2551
- ARM7 LPC2148
- Temperature Sensor LM 35
- Pressure Sensor

- Fuel Sensor

V. SOFTWARE USED

- KEIL µVision
- Flash Programmer
- Proteus 7.2

VI. ADVANTAGES

- Wiring Complexity is less
- CAN Stations can be added or removed easily.
- It allows microcontrollers to communicate with each other.
- It provides high speed and real time communication.
- Provide noisy immunity in an electrically noisy environment.
- It is of low cost and reliable.

VII. APPLICATIONS

In Avionics: CAN communication replaces the 90 wires from just one control panel in avionics compartment by only 2 wires thereby reducing the unnecessary aircraft weight.

In Industry: To interconnect process control units, machines and production subsystem

In Building Automation: To manage air ventilation, heating, doors, elevators, escalator and lighting.

In Agriculture: For control of equipments and for parameter settings.

In Entertainment: CAN bus is used to recreate real-life racing laps in the videogame using the game's GPS Data Logger function, which allow players to race against real laps.

VIII. CONCLUSION

This paper presents a brief overview of CAN protocol. CAN bus is suitable for application requiring large number of short messages with high reliability in rugged operating environment. CSMA/CD allows every node on the bus to have an equal chance to yield access to the bus. This protocol is well suited when data is needed by more than one location because it is message based and not address based. Bandwidth will be always available for critical messages to be transmitted because faulty nodes will be automatically dropped off by the CAN.

ABOUT THE AUTHOUR

S.Aravindh completed his Bachelor Degree in Computer Science and Engineering from Madurai Kamarajar University of Madurai as in the year 2004. Master of Business & Administration from Anna University in the year 2009. Master of Technology in Computer Science and Engineering from Bharath University Year 2011.Presently Working as Associate Professor and Head of the Department of Computer Science and Engineering for Gojan School of Business and Technology Chennai, Published Six International & National Journals and attended Five International Conference Conducted by Reputed Colleges.Area of Interest are Computer Networks, Android Applications, Robotics, Digital Imaging, Optimization Techniques, Soft Computing, Optimization Techniques and Pattern Recognition.

REFERENCES

[1] **"Automobile Control System using Controller Area Network"**, International Journal of Computer Applications 67(18):34-38, April 2013

[2] R. Davis, N. Navet, **"Traffic Shaping to Reduce Jitter in Controller Area Network (CAN)"**, ACM SIGBED Review, Special issue of the 24th Euromicro Conference on Real-Time Systems (ECRTS12),Volume 9, Issue 4, pp37-40, November 2012.

[3] Shuqing Guo; Coll. of Transportion-Engerring, Beihua Univ., Jilin, China "**The application of CAN-bus technology in the vehicle**" 755 – 758, Aug. 2011.

[4] Yujia Wang, Hao Su, Mingjun Zhang., **"CAN-Bus-Based Communication System Research for Modular underwater Vehicle"**, 2011 IEEE DOI 10.1109/ICICTA.

[5] Guo-sheng Feng ; Shijiazhuang Railway Inst., Shijiazhuang, China ; Wei Zhang ; Su-me Jia ; Han-sheng Wu **"CAN Bus Application in Automotive Network Control"** Volume:1 , 779 – 782, March 2010.

[6] .Prodanov, W., M. Valle, and R. Buzas, "**A controller area network bus transceiver behavioral model for network design and simulation"**. IEEE Transactions on Industrial Electronics, 56(9): p. 3762-377, 2009.

[7] Wanke Cao ; Inst. of Vehicle, Northeastern Univ., Shenyang ; Tianxia Zhang ; Yongquan Jia **"Integrated Technology of Vehicle Network Based on CAN"**, Volume:2 , 8301 – 8305,2006.

[8] Scharbarg, J.; IRIT-ENSEEIHT, Toulouse; Boyer, M.; Fraboul, C. **"CAN-Ethernet architectures for real-time applications"** Volume: 2, 8 pp. - 252, Sept. 2005.

[9] Wilfried Voss, "**A comprehensive guide to controller area network"**, Copperhill Media Corporation, 2005-2008.

[10] L. Vhzquez, L. Rojas, E. Uribazo. 2000. **"Intelligent CAN - ISA Gateway Development to Industrial Networks and Internet Coupling"**, IEEE.

[11] Navet, N.; LORIA Comput. Sci. Lab., Nancy, France, **"Controller area network [automotive applications]",** Volume: 17, Issue: 4, 12 – 14, Oct/Nov 1998.