CAN bus Distributed Control and Monitoring System

Abstract: The CAN bus (Controller Area Network) is widely used in different fields of industry and in particular in the car industry,Tthere are a variety of devices, which allow the analysis of data travelling through the bus and generation of messages. The majority of these devices are cards with different formats (PCI, PCMCIA, PXI, etc…) that are finally inserted in a personal computer or rack where data analysis is performed. This article describes a CAN bus monitoring system which allows the analysis and data generation from any PC connected to Ethernet network thanks to the use of a programmable hardware interface which collects the information from the CAN bus and transmits it to any machine requesting this data, using the Ethernet network under a TCP-IP connection. The data monitoring system is extended with the use of OPC protocol (Ole for Process Control) as a tool for secure data distribution to the monitoring and registering systems based on OPC clients

II. SYSTEM DESCRIPTION

The distributed control and monitoring system for the CAN bus outlined below allows the different technicians who use the CAN bus, to analyse the selected data circulating through the bus.

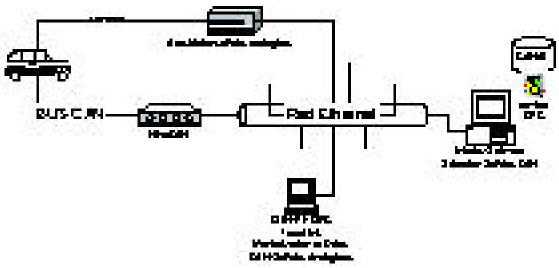

Fig. 13. Block diagram of the control and monitoring system.

1.1 Interface Software

The data to be monitored is normally inside a database (*.DBC), which holds the information related to different messages and signals contained within the messages. (Fig. 15). One of the most widely-used databases in the car industry is CANdb (CAN Data base) from Vector-Informatik. Some of the data related to these messages in the databases are the identifier (Id), frame format and the number of data bytes in the message. Each message contains data from different signals and the database specifies the name of every signal, number of bits of each signal, its position inside the message, the units and information on scale and offset and then converts it into physical units. This system allows the data base and the selection of signals that are going to be filtered for monitoring to be read

Fig 14. Interface Sofware

The program which deals with the selection has been developed using National Instruments graphical language LabVIEW . This interface allows the test technician to select the information to be monitored. This application also allows the periodic generation of several messages for the CAN bus because, in order to take a control action on system devices based on CAN bus, the commands must be sent in a specific format containing specific information.

The interpretation of the database CAN messages is performed without the use of any kind of CAN interface installed in the PC since, as we will explain, the data is received through the Ethernet network via TCP-IP.

1.2 Interface Hardware

Once the information to be monitored has been selected, the LabVIEW application links a TCP-IP connection to a programmable hardware interface which acts as a bridge between CAN bus and Ethernet network. This interface hardware is a programmable embedded system based on an embedded Linux Operating System on a Samsung S3C4510B, which incorporates a Philips PCA82C251 Can bus controller that supports the 2.0A and 2.0B protocols. This device can be connected via RS232 to a PC to be programmed and to be connected to an Ethernet network. The name of this interface, developed by EMS Thomas Wünsche, is EtherCAN CI.

In regard to the operation of the CAN-Ethernet interface, it is possible to filter the messages to be sent to the PC via TCP-IP It is also possible to send all the data through the CAN bus and filter them in the LabVIEW application. This second option has been chosen since there is no saturation or overload of the network because of the velocity of the CAN bus of 500kbps and a 100Mbps Ethernet network is used.

The use of this kind of CAN-Ethernet interface provides data analysis in any computer connected to the Ethernet network and it is not necessary to perform the test beside the system to be monitored, in our case a car test.

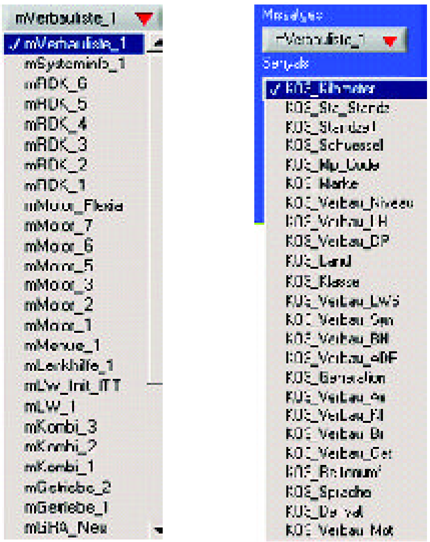

Fig. 15. CANdb messages and signals

1.3 Distribution and Monitoring System

Inside the interface software the use of an OPC data server is included for remote data exportation and monitoring by this communication protocol with any application that can work as an OPC client. The designed system uses an OPC CanaryLabs TrendLink client. Data monitoring is possible from any PC connected to the Ethernet network thereby allowing the analysis of different parameters of the same test performed in different machines by different technicians. To implement the OPC server a DSC toolkit from National Instruments has been used.

III. CONCLUSIONS

A CAN bus control and monitoring system has been designed and implemented. The tests have been carried out in the Technical Center at Seat in the Climate Chambers department. Thanks to this system and the possibility of collecting data from the CAN bus via Ethernet and exporting them using an OPC server,we can now carry out tests where the collected analog data monitoring, by different instruments are integrated into one application with data coming from a CAN bus. We are also able to compare the information in real time as the test is being performed.

Distributed Temperature Measurement System

A distributed and flexible open sensor system is under development which not only allows sensors from different manufacturers to be connected to the same bus (interconnectivity) but also permits their substitution by other manufacturers' equivalent sensors (exchangabality). A sensor network whose hardware and software design is based on Micro Lan (Dallas- Maxim Semiconductor, 1998) utilization which offers components with serial numbers as (DS2401/ DS1990A), Addressable Switch (DS2405),Digital Thermometer (DS1820), A/D Converter (DS2450), Port adapter (DS9097E) and Memories.

The 'MicroLan' presents a master-slave architecture using a single data line and an earthed reference line for communication. It can manage simple low cost sensor interfaces 300m away from master without the need of any special hardware to build the network, using only a telephone cable («Unshielded twisted pair UTP5 100MHz»LUCENT-I SYSTIMAX GIGASPEED 1071A 4/24"). A 16.3kbits/s is more than enough to address a network node and to receive data below every 7ms. With 256 logic addresses, this network beats all standard existing networks, without conflict risks in node addressing because every component manufacturer guarantees only one single node address. The network is not limited to a certain predefined structure where all the components or elements are connected to a backplane and this being connected to a RS-232 communication port of the microprocessor.

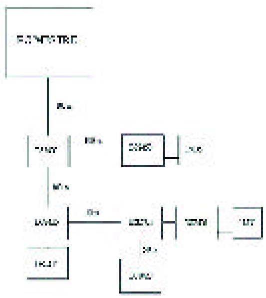

Fig. 16. Block diagram of the implemented network.

The network uses standard voltage levels where 0.8V indicates a low logic level ('LOW') and 2.2V a high logic level ('HIGH') supporting operative voltage levels between 2.8V and 6V. To communicate, all MicroLan elements are powered at the same time through the network. The minimum voltage has to be 2.8V to charge up the 800pF internal capacitor used to store energy where the charge up time depends on the capacitor value and the internal resistance of devices (1kW), cables and contacts and finally cable parasitic capacitor. The necessary waveforms to write commands