INTELLIGENT SENSOR WITH CAN INTERFACE

Assoc. prof. Ph.D. Ratcho Marinov Ivanov, rmi@vmei.acad.bg
Assist. prof. Alexander Stefanov Kerezov, akerezov@vmei.acad.bg
Ivan Blagoev Lesichkov, lesichkov@yahoo.com

TECHNICAL UNIVERSITY OF SOFIA
Faculty of Electronics
Department of Electronics

Summary: *The controller area network (CAN) uses a serial multimaster communication protocol that efficiently supports distributed real-time control, with a very high level of security, and a communication rate of up to 1 Mbps. The CAN bus is ideal for applications operating in noisy and harsh environments, such as in the automotive and other industrial fields that require reliable communication or multiplexed wiring. CAN protocol offers a very good price/performance ratio. Furthermore, the data is very reliable and the error detection is sophisticated and robust. Using advantages of the CAN interface, it is possible to design intelligent sensors with CAN Interface. This paper describes advantages and disadvantages of this type of sensors and building a sensor network with them.*

1. INTRODUCTION

The controller area network (CAN) uses a serial multimaster communication protocol that efficiently supports distributed real-time control, with a very high level of security, and a communication rate of up to 1 Mbps. The CAN bus is ideal for applications operating in noisy and harsh environments, such as in the automotive and other industrial fields that require reliable communication or multiplexed wiring. Prioritized messages of up to eight bytes in data length can be sent on a multimaster serial bus using an arbitration protocol and an error-detection mechanism for a high level of data integrity.

CAN protocol provides advantages over other communication protocols. CAN protocol offers a very good price/performance ratio. Furthermore, the data is very reliable and the error detection is sophisticated and robust.

CAN protocol does not address nodes with physical addresses but instead sends messages with an identifier that can be recognized by the different nodes. This identifier has two functions: it is used both for message filtering and for determining message priority. The ID determines if a transmitted message will be received by any particular CAN module and also determines the priority of the message when two or more nodes want to transmit at the same time.

Sensors are devices, which can measured the parameters of the environment (temperature, pH, light, etc.). Most of them are for low-voltage applications include Value-to-Voltage, Value-to-Frequency and Value-to-Duty_Cycle converters. The

Value-to-Voltage sensors provide a linear output voltage signal proportional to the measured value. The output of Value-to-Frequency converters is a square wave with a frequency that is linearly proportional to the measured value. The output of Value-to- Duty_Cycle converters is a square wave with a duty cycle that is linearly proportional to the measured value. Microcontrollers used those devices to collects information. It is very important, microcontroller to receive the correct values from the sensor. When the sensor is away from the microcontroller, there are many problems, especially when there is a lot of noise. We can solve some of the problems by using the advantages of the CAN interface.

It is possible to design an intelligent sensor with CAN Interface. The sensor should have four pins (two for power supply and two for CAN Bus). The Intelligent Sensor with CAN Interface is shown in Figure 1.

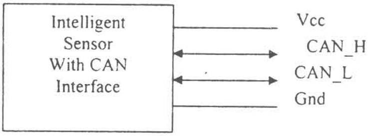

Figure 1 – Intelligent Sensor with CAN Interfaces

2. INTELLIGENT SENSOR WITH CAN INTERFACE

Let the sensor be a temperature sensor. To realize this type of intelligent sensor, this scheme is necessary to contain those functional blocks:

- Temperature sensor
- Microcontroller
- CAN controller
- Transceiver

Block scheme of intelligent temperature sensor is shown in Figure 2.

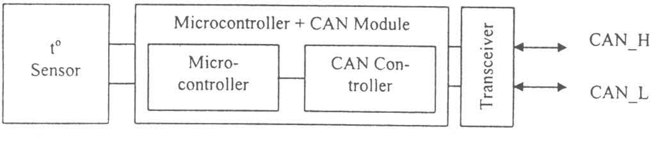

Figure 2 – Block Diagram of the sensor

The task of the microcontroller is to get the information from the temperature sensor, to calculate it and when there is a request from the CAN bus to transmit the temperature.

For the temperature sensor is used SMARTEC temperature sensor - SMT16030. It can measure temperature in the range from $-45^{o}C$ to $+130^{o}C$. It is a 3-pin sensor: 2 for power supply and 1 for pulse output. This sensor generates pulse signal with duty cycle, which is function of the measured temperature.

This pulse signal can be measured with any microcontroller, which has Capture function. After measuring the duty cycle, the microcontroller calculate the temperature according to the formula given from the manufacture.

Connection with the CAN bus is made by using a CAN controller (Stand Alone CAN Controller). There can be used and a microcontroller with on-chip CAN Module.

The CAN protocol supports four different frame types for communication:):

- *Data Frame* – to transfer data. It carries data from a transmitting CAN node to one or more receiving CAN nodes;
- *Remote Frame* – request for data. A CAN Node, acting as a receiver for the certain information, may initiate the transmission of the respective data by transmitting a Remote Frame to the network, addressing the data source via the identifier;
- *Error Frame* – globally signal a (locally) detected error condition;
- *Overload Frame* – to extend delay time of subsequent frames.

The data is very reliable and the error detection is sophisticated and robust.

The implemented CAN controller can detect those types of errors:

- *Bit error* – When the node is transmitting it monitors the bus on a bit-by-bit basis. If the bit level monitored is different from the transmitted one, a Bit error is signaled;
- *Stuff error;*
- *CRC error;*
- *Form error;*
- *Acknowledgment error* – this is detected by a transmitter whenever it does not monitor a dominant bit during the acknowledge slot.

Global errors, which occur at all nodes, are 100% detected. For local errors, i.e. for errors occurring at some nodes only, the shortened BCH coded, extended by a parity check, has the following error detection capabilities:

- Up to five single bit errors are 100% detected, even if they are distributed randomly within the code;
- All single bit errors are detected if their total number (within the code) is odd;

- The residual error probability of the CRC check amounts to 3×10^{-5}. As an error may be detected not only by CRC check but also by other detection process, the residual error probably is several magnitudes less than 3×10^{-5} for undetected errors.

All CAN nodes contain a Transmit Error Counter and a Receive Error Counter, which registers error during the transmission and the receptions of messages, respectively. If a message is transmitted or received correctly, the count is decreased. In case of an error, the counter is increased. The Error Counters have a non-proportional method of containing - an error causes a larger counter increase than a correctly transmitted/received message causes the count to decrease. If one of the Error Counters exceeds the Warning Limit of 96 error points, indicating a significant accumulation of error conditions, this is signaled by the Error Interrupt (error status). A sensor operates in *error-active* mode until it exceeds 127 error points on one of its Error Counters. At this point, it will enter the *error-passive* state. A transmit error which exceeds 255 error points results in the sensor entering the *Bus-Off* state.

The microcontroller must be informed when there are long lasting disturbances and when bus activities have returned to the normal operation. During long lasting disturbances, a sensor enters the bus-off state. Minor disturbances of bus activities will not affect a node. The node does not enter the bus-off state or inform the microcontroller of a short-lasting bus disturbance.

The node, being nearest to the error-locations, react with a high probability, the quickest (i.e. becomes error-passive or bus-off), hence errors can be localized and their influence on normal bus activities is minimized.

All of this advanced error handling is done automatically by the CAN controller, without any need for the host microcontroller to do anything. This is one of the big advantages of CAN.

CAN transmits signals on the CAN bus which consists of two wires, a CAN_High and CAN_Low. These 2 wires are operating in differential mode, that is they are carrying inverted voltages (to decrease noise interference) The voltage levels, as well as other characteristics of the physical layer, depend on which standard is being used. The CAN bus is made with a twisted pair. The transmission rate depends on the bus length. For a bus smaller than 40 meters, the transmission rate is up to 1 Mbit/s. For the Transmitter can be used MTC-3054, L9615, MCP2505, MC33388, PCA82C250, TJA1050, SN75LBC031 or other CAN transceivers.

The most basic algorithm, which is executing by the microcontroller, is the follow: The microcontroller permanently measured the duty cycle of the pulses from the sensor, calculate the temperature and store the result in the variable "Temperature". When there is a request from the CAN Host node, the microcontroller sends the last actualization of the variable "Temperature". In this way, the Host node receives the temperature in suitable type. It is not necessary to make additional calculations.

Additional capabilities, which can be added to this algorithm, are:

- to calculate the temperature in the measured unit, which is used by the host node (°C, °K, °F etc.);
- to transmit the value on every measurement, if the Host node send a command for continuous transmitting;
- to inform the Host node if the temperature is equal to the value, which is defined before;
- to make a statistic calculations.

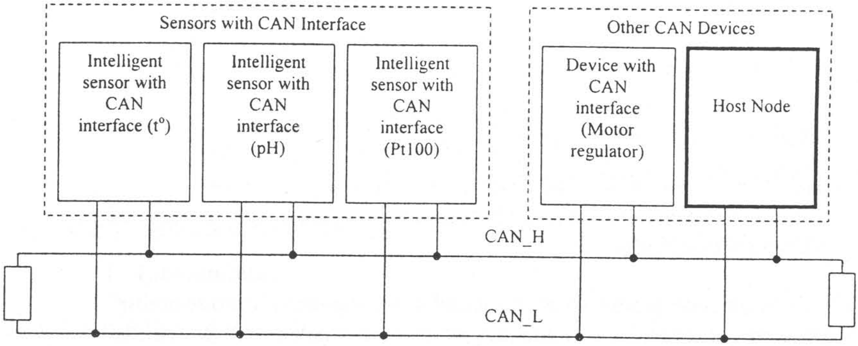

Figure 3 – CAN Network

3. CONCLUSIONS:

Advantages of the intelligent sensor with CAN Interface:

- There is no need the host node to measure the duty cycle and to make those taking many machine time calculations (floating point calculations);
- A sensors network can be built by using those types of sensors. Host node can very easy collect the information from the sensors by using the corresponding commands;
- All the sensors can be linked by using only 2 wires.

Disadvantages of the intelligent sensor with CAN Interface:

- There is need of an additional hardware (Microcontroller, CAN Controller and Transceiver). But if the sensor is produced in a big amount the cost of the sensor will not be changed significantly;
- Due to the practical limitation of the hardware (transceivers), it can only link up to110 nodes (with 82C250, Philips) on a single network. But networks with more then 110 node are not wildly distributed.

REFERENCES:

[1] Monnet, Claire. "*Understanding the CAN Controller on the TMS320C24x DSP Controller.*" Texas Instruments Literature Number SPRA500, December 1998.

[2] Wolfhard Lawrenz, "*Worldwide status of CAN – Present and future*", Second International CAN Conference – London, 1995.

[3] Bhargav P. Upender, Alexander G. Dean, "*Variability of CAN Network Performance*", Third International CAN Conference – Paris, 1996.

[4] Steffen Rietz, Wolf-Joachim Fischer, "*Multi sensor controller with CAN field bus Interface*", Forth International CAN Conference – Berlin, 1997

[5] *DeviceNet Europe - A Centre for Controller Area Network and DeviceNet*, http://www.warwick.ac.uk/devicenet/

[6] *CAN*, http://www.goofee.com/can.htm

[7] *CAN Controller Area Network Information,* http://www.algonet.se/~staffann/developer/CAN.htm