Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Engineering 38 (2012) 791 – 798

Procedia Engineering

www.elsevier.com/locate/procedia

International Conference on Modelling, Optimisation and Computing (ICMOC 2012)

Portable Embedded Data Display and Control Unit using CAN Bus

Nanda kumar G[a*],Raj Narain B[b]

[a,b]*Department of Electronics and Instrumentation,Velammal engineering college,Chennai-600066,India*

Abstract

The development trend and research focus of control system is networked control. The Field Bus Control System (FCS) which is important segment in networked system is widely applied in control system. The CAN (Controller Area Network) bus which has strong fault tolerance, high reliability and low cost plays an important role in FCS. The CAN is a serial communication protocol which efficiently supports distributed real-time control with a very high level of security. Its domain of application ranges from high speed networks to low cost multiplex wiring. In automotive electronics, engine control units, sensors, anti-skid-systems, etc. are connected using CAN with bit rates up to 1M bit/s. This project is to develop an automatic monitor and control of any non- industrial process using a Portable Embedded data display and control unit based on CAN Bus. It is implemented by applying 16 bit RISC key microprocessor technique, embedded software technology, embedded GUI (Graphics User Interface) technology, CAN bus communication technology, information storage and management technology[6]. The Proposed system involves the CAN Protocol to communicate the data. The system consists of many nodes each comprising a PIC microcontroller and a sensor. The data is retrieved from the sensor and forwarded to a PIC microcontroller through a CAN transceiver. The PIC Microcontroller establishes connection with a CAN Protocol and process the sensor data and the data through the CAN Bus which is twisted pair of wires. Another PIC microcontroller at remote end is interfaced with LCD (Liquid Crystal Display) to view the sensor data and send back control signal to the sensor through CAN protocol. The research and development of this project was driven by robustness of CAN communication and its practicality of application.

Keywords: CAN (Control Area Network);PIC (Peripheral Interface Controller);nodes;sensors;.MCP2551 transceiver;LCD (Liquid crystal Display)

* Corresponding author. Tel.: +91 9884747404.
E-mail address: ng_nandakumar@yahoo.in.

doi:10.1016/j.proeng.2012.06.099

1. Introduction

The origin of the field bus was to replace any point-to-point links between the field devices (Field Devices are simply the Sensors and Actuators of the plant) and their controllers (like PLC's, CNC's etc.) by a digital single link on which all the information is transmitted serially and multiplexed in time. Choosing the serial transmission has many merits in comparison with other kinds of transmission like parallel transmission. For instance, the sequential or serial transmission reduces the total required number of the connecting lines over greater distances than that of the point-to-point or even parallel transmissions. It is because of this advantage the field buses are mostly preferred in industries for both process control and manufacturing [3]. Also field buses play a major role in the modern industry automation development. This is not everything, but it takes a prominent place in interconnecting advanced real-time distributed systems that are used in avionics, data acquisition, and the modern automotive applications.

There are two types of communication system used in the security and control system of industries which are Field bus Control System and Distributed Control System [2]. Among these two systems, the Field bus control system is a solution of process control with advantages in standardization in network communication. It has simpler structure, higher accuracy and higher anti disturbance capability than the distributed control system. For a large number of components, field bus control system has more advantage compared to distributed control system. Controller Area Network (CAN) bus is one of the Field bus control system type used in decentralization, intelligence and networking.

CAN protocol have been designed by Robert Bosch in 1986 for automotive applications as a method for enabling robust serial communication. The goal was to make automobiles more reliable, safe and fuel-efficient while decreasing wiring harness weight and complexity. Having all these advantages, the use of CAN bus in a monitoring system is an added value to the system and increase its reliability.

The purpose of using CAN bus is to enable any station to communicate with other station without putting too much load to the main controller [2]. This opens the opportunity in development of intelligent ubiquitous sensor network and controller system.

CAN bus is a fast serial bus that is designed to provide an efficient, reliable and economical link between various CAN stations, sensors and actuators. The requirement for the information exchange has grown to such an extent that a cable network with a length of up to several miles and many connectors were required [1]. This will increase the problems concerning material cost, production time and reliability.

This paper will explain the overview of CAN bus system, design method of a portable handheld unit using CAN Bus protocol, system hardware, software and the result.

2. Project Objective

The aim of this project work is to develop an Automation application of embedded data display and control terminal based on CAN bus protocol. Therefore the completion of the project is expected to fulfil the following objectives:

- To develop a cost effective automated embedded system,
- To apply CAN bus system in data acquisition and control,
- To build a model of the system utilizing the CAN protocol,
- To design and develop a multiple node portable monitoring and control system.

3. CAN Protocol

The CAN protocol is based on a bus topology, and only two wires are needed for communication over a CAN bus. The bus has a multi master structure where each device on the bus can send or receive data [1]. Only one device can send data at any time while all the others listen. If two or more devices attempt to send data at the same time, the one with the highest priority is allowed to send its data while the others return to receive mode.

The CAN communication protocol is a CSMA/CD protocol. The CSMA stands for Carrier Sense Multiple Access, which means that every node (device) on the network must monitor the bus for a period of no activity before trying to send a message on the bus (Carrier Sense). Also, once this period of no activity occurs, every node on the bus has an equal opportunity to transmit a message (Multiple Access). The CD stands for Collision Detection. If two nodes on the network start transmitting at the same time, the nodes will detect the 'collision' and take the appropriate action based on the priority of nodes [3]. The figure1 below shows the example of CAN Bus communication.

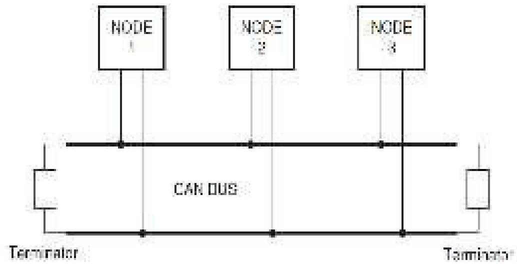

Fig.1. CAN Bus Architecture

4. Hardware

This portable system consists of multiple nodes with each node having a PIC microcontroller, CAN transceiver, sensor and devices.

4.1. PIC18F458 Microcontroller

The microcontroller used in this work is Peripheral Interface Controller popularly known as PIC microcontroller. Among the various microcontrollers such as 8051, ARM, Arduino etc., the PIC microcontroller is used because of the following advantages.

- Reduced instruction set
- Easy availability
- Low cost
- High operating frequency and performance
- Availability of many ports for sensor interface
- In-built CAN controller.

The PIC microcontroller used in this work is PIC18F458. It is selected among various microcontrollers because it has a built- in CAN controller as the result of which the number of devices used is reduced and hence the space requirement.

4.2. CAN Transceiver

The CAN Transceiver used here is MCP2551. This chip supports 1 M bits/s data rate which can connect up to 112 nodes. It act as an interface between the CAN protocol handler (PIC18F458) and CAN bus, by converting the digital signals generated by the CAN protocol handler into differential signals which will be transmitted over the CAN bus via the CANH and CANL pins of MCP2551.

4.3. Sensors and Devices

The data is retrieved from the sensors connected at each node. Sensors can be selected as per the requirement and the area of application. Since this work is intended to be used for a non industrial environment sensors used here is temperature sensor, smoke sensor, IR sensor. Additional devices such as LCD, keypad and other devices can be used which are controlled by the microcontroller if the sensor values exceed the pre-set value programmed earlier.

This project work is focused in designing a CAN bus system module at a discrete component level where all circuitry are designed and fabricated. Each of the module are similar except the application of such are different based on the sensors used. The Generalized model block diagram of the project is shown below in figure 2.

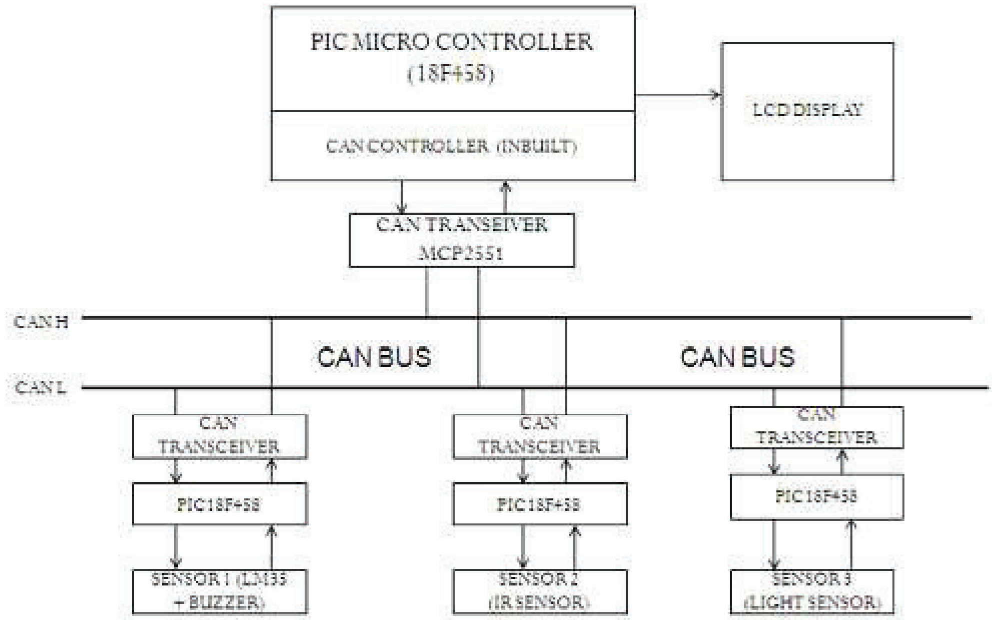

Fig. 2. Block diagram

The proposed system consists of a main controller which is interfaced with the CAN transceiver and LCD. This controller is used to receive data from the CAN bus and display the status of individual sensors on GUI and also send signal back to the bus to operate other devices. Besides the main controller the design consists of integration of many sub-systems as shown. Each sub system (node) consists of the PIC18F458 microcontroller, CAN transceiver and sensor or an electrical device.

The design also includes a battery set up to power up the microcontroller and sensor. The microcontroller requires a +5V supply and the sensor requires voltage between +5V to +9V rated at 1A maximum. So the overall set up requires a +9V battery for each node. The nodes shown here are placed at different places and the main controller is placed at a remote place from where the monitoring can be done. Consider a two node system of the above system with a main controller and a node with the temperature sensor. The circuit connection is made as shown in the figure 3 below and the working is explained as follows

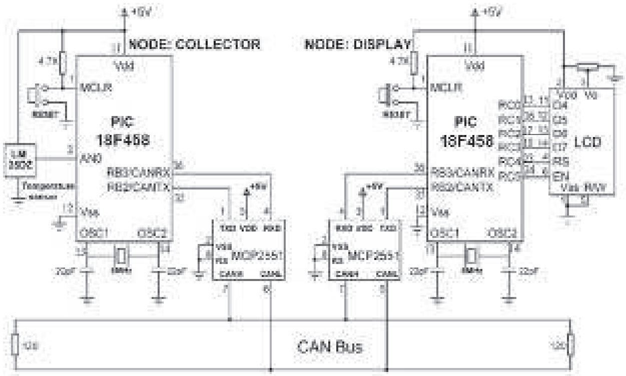

Fig. 3. Two Node System

It consists of two nodes namely display node and collector node. The collector node is used to collect the data from a semi-conductor type temperature sensor (LM 35). The sensor can measure temperature in the range of 0°C to 100°C and generates an analog voltage directly proportional to the measured temperature (i.e., the output is 10mV/°C). For example, at 20°C the output voltage is 200mV. The CAN outputs (RB2/CANTX and RB3/CANRX) of the microcontroller are connected to the TXD and RXD inputs of an MCP2551-type CAN transceiver chip. The CANH and CANL outputs of this chip are connected directly to a twisted cable terminating at the CAN bus.

Like the Collector node, the display node also consists of CAN transceiver and PIC18F458 microcontroller. This node receives the temperature data from the differential voltage CAN bus using CAN transceiver MCP2551 and the PIC18F458 microcontroller converts the voltage back into temperature value and displays it on the LCD. This process is repeated continuously with preset time delay. All the other nodes functions in the same way as described above with the change lying only in the type of sensor used.

5. Software

In order for the hardware to function, the firmware code for the system is to be written. Theoretically, software that resides in the non-volatile memory and handles the operation as well as the function of the system is known as firmware. The firmware holds the information that a microcontroller needed to operate or run. Thus it needs to be free of errors or bugs for the successful operation of the end product. There are various types of software that could be used to program the PIC microcontroller. Program can be written in either in C, Basic or Pascal or even Assembler.

In this project work the program is written in C language using either Mikro C or CCS compiler to generate the hex file. This hex file is then downloaded into the PIC microcontroller for it to function as programmed. There are several advantages in using Mikro C or CCS compiler as it is built with library function for CAN bus. This reduces the time to write the program code as well as to ease the development process. The main challenge of developing the program lies in the setting of timing parameters among the nodes so that they work in synchronized condition without errors. The program flow in collector node and display node is shown in the figure 4 below.

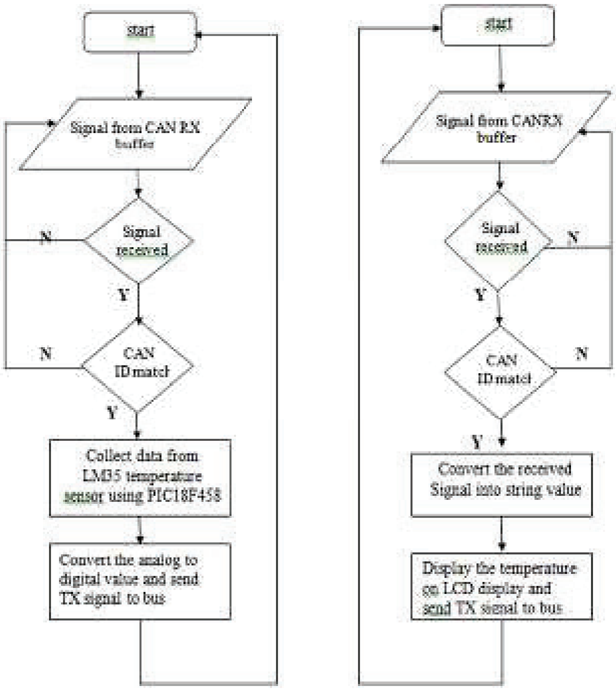

Fig. 4. Flow chart of program sequence in collector node and display node

6. Data Transfer in CAN Bus

The data transfer along the CAN bus is the important criteria in CAN communication protocol. The message is sent along the bus in the form of Data frame. The data frame may be either 11 bit standard format or 29 bit extended format. The standard 11 bit identifier frame format is shown in the figure 5 below which has a start of frame (SOF), arbitration field, control field, data field, CRC field, acknowledgement field, End of frame.

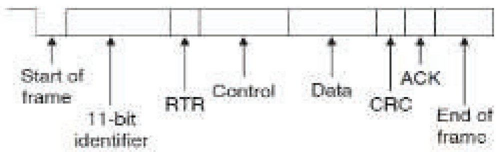

Fig. 5. Data Frame Format

- Start of frame: The start of frame field indicates the beginning of a data frame and is common to both standard and extended formats.
- 11- Bit identifier: This field is an 11 bit field which gives priority to the device using arbitration method.
- RTR field: Remote Transmission Request (RTR) is used to request for message from a remote device.
- Control field: The control field is 6 bits wide, consisting of 2 reserved bits and 4 data length code (DLC) bits, and indicates the number of data bytes in the message being transmitted.
- Data field: The data field carries the actual content of the message. The data size can vary from 0 to 8 bytes. The data is transmitted with the MSB first.
- CRC field: Cyclic Redundancy Check consisting of a 15-bit CRC sequence and a 1-bit CRC delimiter is used to check the frame for a transmission error. The CRC calculation includes the start of frame, arbitration field, control field, and data field. The calculated CRC and the received CRC sequence are compared, and if they do not match, an error is assumed.
- ACK field: The Acknowledgement field indicates that the frame has been received normally. This field consists of 2 bits, one for ACK slot and one for ACK delimiter.

This data frame explained above is used to assign priority to the nodes connected on the CAN bus based on arbitration technique. Arbitration is used to resolve bus conflicts that occur when several devices start sending messages on the bus at a same time. The arbitration field indicates the priority of a frame, and it is different in the standard and extended formats. In the standard format there are 11 bits, and up to 2032 IDs can be set. The extended format ID consists of 11 base IDs plus 18 extended IDs. Upto 2032 x 10^18 discrete IDs can be set.

During the arbitration phase, each transmitting device transmits its identifier and compares it with the level on the bus. If the levels are equal, the device continues to transmit. If the device detects a dominant level on the bus while it is trying to transmit a recessive level, it quits transmitting and becomes a receiving device. After arbitration only one transmitter is left on the bus, and this transmitter continues to send its control field, data field, and other data. In this way the data transfer is carried out in the CAN line.

7. Result and Voltage Level On The Can Bus

The health of the physical layer of a CAN bus is important. The robustness of the CAN physical layer such as ISO 11898-2 cover up many electrical problems such as open, shorted, leaky, out-of-spec or unbalanced lines. The network traffic on many buses, including CAN, is not repetitive and therefore rather difficult for a standard oscilloscope to reliably trigger on and display. Digital storage oscilloscopes are needed to effectively view these waveforms. Ordinary oscilloscopes rely on displaying the same image repeatedly and this requires a stable and repetitive signal for a clear and jitter-free display. Figure 6 shows the voltage output of CAN signal. This is done by taking advantage of the fact that the grounds of the oscilloscope and of the CAN system are not usually connected together. Connect the oscilloscope hot lead to one CAN line and the ground to the other CAN lead. The differential waveform will display correctly.

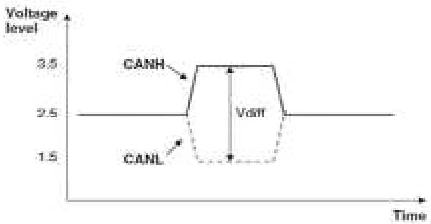

Fig. 6. Voltage Level on the CAN bus

If the voltage level on CAN bus between CANH and CANL is more than 2.5V it implies that data is being transmitted on the bus. If it is below 2.5V then data is not transmitted through the bus. This can also be tested simply with the help of multi meter, but the voltage is not constant to measure it continuously. Figure 7 shows the two node system implemented on the bread board.

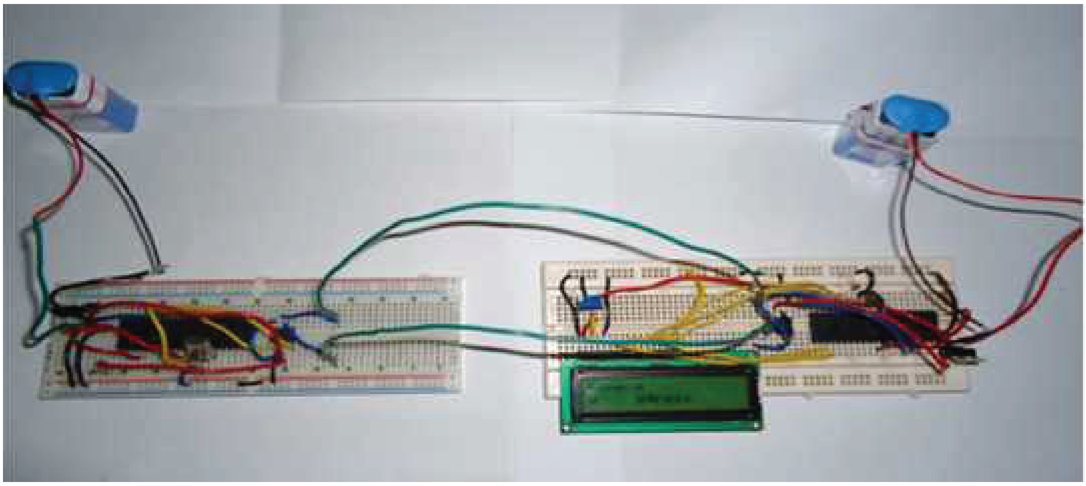

Fig.7 Hardware demonstration of two node system

8. Conclusion

Thus the demonstrator of an embedded data display unit with a two nodes is designed with CAN bus for communication. At the end of this project, the demonstrator had been successfully developed; performing simple application that displays the temperature and imitates the communication over the network. Like this, the CAN bus communication can be fully utilized in the development of an application specific dependable networking system for further work in various fields either for industrial or non industrial purpose in reduced cost.

References

[1]. Dogan Ibrahim, (2008) ‘Advanced PIC Microcontroller Projects in C’ Elsevier Ltd.

[2]. Mazran Esro, Amat Amir Basari, Siva Kumar S, (2009) ‘Controller Area Network Application in security system’ World Academy of Science, Engineering and Technology vol 59

[3]. Michel Nabil Mikhail Hanna, (2004) ‘Real-Time Analysis of FIP-based Systems’MS thesis, Chapter 1 Introduction, Faculty of engineering,Cairo university

[4]. Milan Verle,(2009) ‘PIC microcontrollers programming in C’Mikroelectronica 1st edition

[5]. Nigel Gardner, (2003) ‘PICmicro MCU C - An Introduction to Programming the Microchip PIC in CCS C’ CCS corporation

[6]. Peng Yang, Le Yang, Xin Guo, (2009) ‘Research on Embedded data display unit based on CAN bus’ 978-1-4244-2800-7/09/$25.00 © IEEE

[7]. Robert Boys (Dearborn Group), (2005) ‘Practical testing of the CAN Physical layer’ CAN Newsletter 2

[8]. www.ccsinfo.com, ‘Quick start with CCS C Compiler’ Custom Computer services, INC

[9]. www.mikroe.com, ‘mikroelectronica development tools’ Mikroelectronika.

[10]. Zarina Binti Mohd. Noh, (2009) ‘Development Of A Dependable Distributed Embedded System’ Universiti tecknologi, Malyasia.