CAN IN AUTOMATION AND MANUFACTURING SYSTEMS
Huseyn Ekiz* Etem Koldakaya** Osman Gurdal***
* Sakarya University, Technical Education Faculty
** Hacettepe University, Engineering Faculty, Adapazari, Turkey
*** Gazi University, Technical Education Faculty, Ankara, Turkey

Abstract

In industrial applications, a network must provide communication between the different microcontrollers or computer-based systems in manufacturing processes such as programmable tools, instruments, control machines. Controller Area Network (CAN) is one of the bus-based systems most suitable for distributed control system operations because of the high data transfer rate and data security. The CAN technology is very useful for any product / system with multiple microcontrollers and general purpose sensor / actuator bus systems for distributed real-time control which could be used in industrial automation / manufacturing. In this study, a CAN based production line is designed. In addition, to be able to investigate the performance of the CAN in industrial environments and to show the applicability of CAN in automation / manufacturing systems, a CAN networked production line for a chemical compound factory is detailed and simulated.

In this paper, firstly, a brief overview about CAN is given. Then, performance analysis of CAN networked production line for chemical compound factory is presented and simulated. The system is simulated not only at single segment but also bridged CAN segments. Lastly, the results obtained from the simulation are given and finally some conclusions resulting from the work are presented.

Key Words: Controller Area Network (CAN), Manufacturing Systems.

1. Introduction

In industrial applications, a network must provide communication between the different microcontrollers or computer based systems in manufacturing process such as programmable tools, instruments, control machines. In addition, the process engineering and petrochemical industries have a considerable need for measurement and control technology [CANM95]. CAN is one of the bus systems most suitable for distributed machine control systems because of the data transmission safety and the high data transfer rate [RAUC94].

The CAN has been developed to provide a serial interface between electronic control units (ECUs) in automotive applications and it has become an industry standard (ISO 11998 and 11519-1). Now it is a desirable, cheap solution for networks in industrial environments [McLA95]. The CAN has been already used in wide range of applications such as textile machines, robotics, measurement systems, drive control systems, PLC controlled manufacturing systems, mobile hydraulic systems, automation control, machinery control, medical systems, even in agricultural applications [KHOH94, LAWR95]. In addition, it has been shown that the CAN technology is very useful for any product / system with multiple microcontrollers and general purpose sensor / actuator bus systems for distributed real-time control which could be used in industrial automation [McLA95].
However, there are only a few studies about CAN performance analysis in industrial automation systems [RAUC94]. So, applicability of the CAN based systems in industrial automation should be investigated. Following, an example of CAN based systems in industrial automation is presented and performance analysis of the CAN in this system is investigated.

1.1. Controller Area Network

The Controller Area Network (CAN) is a serial bus with high speed, high reliability, and low cost for distributed real time control applications. The Controller Area Network communication protocol is a contention-based serial communication. As an access method, the CAN uses CSMA/CD, Carrier Sense Multiple Access with Collision Resolution. The CAN although serial in nature is unlike many serial communication protocols; it contains no information relating to the destination or source addresses. Instead, the message contains an identifier which indicates the type of information contained in the message. The identifier is not only used to identify the message but also used in the arbitration mechanism. The CAN associates a priority with each message to be sent and uses a special arbitration mechanism to ensure that the highest priority message is the one transmitted. In CAN, data is transmitted as a message consisting of between 1 and 8 bytes. A message may be transmitted periodically, sporadically, or on-demand and is sent as a frame (Figure 1). More detail about the CAN protocol and CAN frames can be found in [EKIZ97].

![Figure 1. Standard CAN data frame format](image)

1.2. Evaluation of Electronics Control Units and CAN

In this step, explanation of the evaluation of electronic systems integration in distributed systems will help to understand the importance of the CAN (Figure 2). In the past, an electronic system was designed as independent and self supporting (Figure 2.a). Current systems, however, are interactive in that they share information via hard wire interconnections (Figure 2.b). As these interconnections have increased, the need for serial interfaces and an open bus system (Figure 2.c) is being recognised. Future architectures will have intelligent sensors and actuators that are capable of accessing a serial bus directly (Figure 2.d). Possibly there will be a lower level / lower speed bus dedicated to the sensors and actuator interface [KHOJ94]. This means that, the CAN is a convenient solution for present and future electronics control system's architectures.

2. Modelling a CAN-based Manufacturing System

CAN based manufacturing system to be modelled contains a chemical compound manufacturing situation in which each process is supported by a microcontroller-based machine. In such an industrial application, CAN provides communication between different controlling devices in the system. Each microcontroller-based process (station), which is called a node, executes its own programs and controls frequent transfer of data among stations.
**Figure 2. Evaluation of electronics systems integration in distributed systems**

In the modelled CAN application, each node (microcontroller) is loaded with its main program and a guidance routine to execute the operations. In the system, each node executes segments of their main programs that do not involve any data transfer. When transfer is required, one node sends messages to one or more nodes based on broadcasting principles. This message is received by the nodes and the intended node will invoke the appropriate operations. During the process, each node may request data from another node or nodes by sending a request message. Request messages are executed, depending upon the message and node priority.

The designed manufacturing system consists of four processing parts which communicate with each other: general control, production machines, quality control, and packaging / stocking systems (Figure 3).

The **general control processing** nodes control the sequence of operation in the production chain and the necessary synchronisation of events. A **temperature controller** checks and controls the system / factory temperature at different points. To provide this service, this node reads the data from different temperature sensors at predefined intervals. The **program controller** checks the quality control and production systems. This node sends request messages to related nodes to check their status. The **product control node** inspects the packaging and transfer nodes. These nodes send information after the completion of their processes and the product control node can check the status of the packaging and transfer system. All nodes in this system perform their processes depending upon the arrival of the related messages.

**In the production system, a chemical compound machine produces new compounds from different chemical elements.** This node is used with a personal computer (PC) to quantitatively analyse and to change the amount of elements used. The **transmitters**, **servo valves** and **flowmeter** are used to control the transfer of chemical elements and compounds.
produced. Each of the nodes in this system performs its own process and sends an information message to related nodes (e.g. general control nodes and packaging nodes).

The quality control system is also a CAN based system controlling the process to test or visually inspect products, rejecting those that are inferior and passing the good ones onto the next machine in the work cell prior to packaging and transfer. For example, the input buffering module buffers products in a holding area before they are inspected. The offloading module places products after inspection on the two transport bands, depending on the inspection results. The X-Y table places products to be monitored. The processes time of the nodes are represented by time delay in the simulation.

The packaging/stocking system places products which has passed from quality control. The products are counted, then placed, carried, and labelled by the labelling node. The nodes in this system perform the processes based on product input and all nodes send messages to related nodes after performing their processes.

In this application, each node (microcontroller) is loaded with its main program which performs the main job and a guidance routine to execute the operations. During the normal operation, each node executes segments of its main program that does not involve any data transfer. When data transfer or control is required, one node sends messages to one or more nodes based on the broadcasting principles. This message is received by the nodes and the intended node will invoke the appropriate operations. During the operation, each node may request data from another node or nodes by sending a request message. In the system, each node has a different priority, for example the program controller node has higher priority than the labelling node, in the designed system.

![General block diagram of designed and simulated system](image)

**Figure 3. General block diagram of designed and simulated system**

3. Simulation environment

The basic system used for the simulation is shown in Figure 3. A commercial simulation package (Network II.5) that has been developed to simulate network systems is used to obtain the result presented. During the simulation, some of the requests generated by each node are exponentially distributed and others are dependent each other. The mean response time is defined as the delay between a node generating a request message and the node receiving the corresponding response message from the involved node. To present the system and results, a single bus system is used as a basis for comparison with other models to be investigated later. Modelling the system as a single segment with 12 nodes is referred to as Module 1 in the presentation of the results. It should be noted that the important subject to be investigated here is the message transfer among the nodes on the CAN system. In other words, the proposed system is designed to create message traffic on a
CAN system and to show that this CAN system can be used in proposed industrial applications.

Figure 4. Modelling of the manufacturing system.

In the simulation of the system, discrete time and discrete event techniques are used to create realistic message traffic. This means that each node executes two types of processes:

i- To perform some of the processes, a clock is advanced in steps. At each step time the blocks involved are executed. This is known as discrete time simulation. For example, the temperature controller sends a message to the program controller and the chemical compound machine in predefined intervals, or the program controller sends request messages to some nodes to check the system status.

ii- Some of the processes are executed at the occurrence of events such as the arrival of some frames. This is known as discrete event simulation. Most of the message transfers are executed based on the discrete event simulation technique. For example, the XY position table performs its process after receiving a message from the Off-loading module or the labelling node executes blocks after being informed by the placing product module. In discrete event simulation, a common method to create the activities in the model is through random number generation. Generation of random numbers is obtained using exponential or random step distributions. These are called as system message iteration time.

4. Simulation Results of the single segment CAN based Manufacturing System

Figure 5 illustrates the number of messages transferred and bus utilisation of a single segment CAN network in the simulated system. Message iteration time represents the average value of the statistical distribution that is used to create messages in the CAN nodes. It can be seen from Figure 5 that with a message traffic of about 5000 messages/second results in nearly 70% bus utilisation on the CAN bus which is satisfactory to provide a service to measure the hardware utilisation and software execution of the designed system. Bus utilisation of 70% is acceptable according to accepted utilisation limits in network systems [SCHO78].
Figure 5. Number of transferred messages per second and bus utilisation in Module 1

Figure 6. Mean request delay and number of queued message in Module 1

The time period between the time a node's request to access the CAN bus and the time at which the node access is granted is called the request delay. Mean request delay refers to the average of all waiting times during the simulation. The mean request delay is also investigated and results are shown in Figure 6. The mean request delay of the modelled system reaches 0.5 millisecond in high message traffic. This mean request delay is acceptable according to real time requirements which are detailed in [BABA96, TIND94]. According to these papers, the message arrival time of the most critical messages should be less than 5 milliseconds.
If the request delay time is too long, then this results in message queuing waiting to be processed. Figure 6 illustrates the number of queued messages in the modelled single segment CAN system. As seen from Figure 6, the single segment manufacturing system has a maximum of 12 queued messages. Since a CAN chip (e.g., 82527 CAN chip) can store up to 15 message in its message objects, the designed system can be supported using standard CAN elements.

5. Interconnection of Multiple Segment CAN Systems using Bridges

As the application of CAN increases, in terms of the size of area to be distributed there will be problems which will have to be solved. With a CAN bus, limitation of physical length of the bus is 2 Kms, at the speed of 20 kBit/s, but it can be maximum 50m at 1Mbit/sec and 100m at 500Kbit/sec (CANM95b). The appropriate solution of this problem is to segment the system into parts and then, to interconnect them using coupling devices. The CAN segments, each supporting a workgroup of nodes, can be interconnected using bridges and various network topologies [EKLZ96a]. To investigate the behaviour of the CAN systems in such cases, the designed manufacturing system was segmented and then interconnected using interconnection devices in the different network topologies to compare their operation and behaviour. The benefit of this solution is not only to create an industrial application of the CAN system, but also to provide a solution to extend the volume of the application area. Here, only the simulation results of the two segmented CAN system is presented. Results of the other interconnection topologies can be found in [EKLZ96a].

Bridges are used extensively in most large LAN installations. This means that a bridge can be used to extend the length of CAN systems. Bridges receive and buffer all frames in their entirety before performing the relaying function [HALS96]. To plan for an interconnection of two or more segments through the use of a bridge, reasonable network traffic should be developed between nodes and segments [EKLZ96b]. To do so, an attempt should be made to classify nodes (stations) into groups based on the type of general activity performed, and then estimate the network activity for one node per group. Adding up the activity of all nodes and all groups (segments) provides the traffic activity for the entire system. To obtain the characteristics of the designed CAN system as grouped nodes, the designed single segment manufacturing system is segmented (into two segments) and then interconnected using a bridge as shown in Figure 7 (called Module 2).

![Bridged CAN based networking systems-module 2 (M2)](image)

**Figure 7.** Bridged CAN based networking systems-module 2 (M2)

5.1. Simulation results of the two segmented CAN system

To present the systems and results, it may be helpful first to illustrate how transfer requests of single segment CAN network and a basic bridged CAN network (two segments) compare.

The number of transferred messages in the designed system is defined by the tasks to be performed. These tasks can be classified according to the time in which the tasks are produced. The production of a task can be periodic, sporadic and on demand. Periodic tasks are the tasks which begin their execution at regular time intervals. One example might be...
the temperature control system. Sporadic tasks begin their execution at irregular intervals such as program controller requests. Responding to a request from any node is called an on demand task (most of the tasks in the designed system perform processes on demand). The number of messages in Figure 8 represents the messages transmitted on buses in Module 1 and Module 2. System message iteration time illustrates the time which sporadic messages are generated. Sporadic messages in the designed system are generated with an exponential distribution which has an upper and a lower limit. System message iteration time represents the average of the generated numbers as microseconds.

The bus utilisation of the bridged system is stable and lower than that of the basic system since the bus speed is high in the bridged system. The utilisation of the single segment module increases rapidly, while the increasing load is shared by two segments in the bridged system. Utilisation of buses is defined by message transfer time, node acquisition and release times, acknowledgement time, etc. In addition, the utilisation of the buses are related to the number of transferred messages on those buses. Therefore, to decrease the number of messages, the bridge should be located in an appropriate location (to provide maximum local / minimum remote message transfer ratio).

Figure 8. Comparison of the single segment CAN and the bridged CAN systems at the point of number of transferred messages in per second and utilisation of buses.

![Comparison of single segment CAN and bridged CAN systems](image-url)
Figure 9. Average (mean) request delay and queued message number in M1 and M2

Mean request delay of the messages is defined by the message transfer time, bus utilisation, number of transferred message, etc. One of the main parameters among these is the message transfer time which is calculated as follows for the CAN bus (for a maximum length message-8 byte data):

- **Basic cycle time**: 1 microsecond
- **Bits per cycle**: 1
- **Cycle per word**: 8 (8 Bits/word)
- **Word per block**: 8 (8 Words/Block)
- **Word overhead time**: 0
- **Block overhead time**: 47 microsecond (44 overhead + 3 interframe space)

\[
\text{Number of cycle} = \text{bits to transmit} / \text{bits per cycle} = 108 / 1 = 108 \text{ cycle}
\]

\[
\text{Number of words} = \text{number of cycle} / \text{cycle per word} = 64 / 8 = 8 \text{ words}
\]

\[
\text{Number of blocks} = \text{number of blocks} / \text{words per block} = 8 / 8 = 1 \text{ block}
\]

\[
\text{Transmission time} = \left( \frac{\text{number of blocks} \times \text{blocks overhead time}}{} + \left( \frac{\text{number of words} \times \text{words overhead time}}{} \right) + \left( \frac{\text{number of cycle} \times \text{bus cycle time}}{} \right) \right) = (1 \times 47) + (8 \times 0) + (64 \times 1) = 47 + 0 + 64 = 111 \text{ microsecond}.
\]

The bridged system provides a lower request delay and a smaller number of queued messages, since this module has lower utilisation (Figure 9). The lower request delay and smaller number of queued messages result in better performance for the system. A long request delay results in a queue of messages to be transmitted. Queued messages creates a buffer requirement to store the messages in the queue. According to results presented in Figure 9, the bridged topology requires 6 message objects to be stored in the CAN chips, and most CAN chips can provide this facility.

6. Conclusion

In this study, a chemical compound production line was proposed and detailed as an example of CAN based systems. The proposed system was simulated to give an idea about the applicability of the CAN based systems in industrial automation and to investigate the behavior of the CAN in segmented/bridged systems. The proposed situation shows that the CAN technology is very useful for any product/system with multiple microcontrollers and general purpose sensor/actuator bus system in industrial automation. It should note that the scenario used in the designed system based on hypothetical situations. However, the important point is the number of message traffic in per second. This means that the scenario can change but number of transferred message per second will be same.

In the paper, the simulation and performance analysis of large bridged CAN systems have been presented. Single segment and bridged systems were compared using the number of transferred messages, the bus utilisation, and maximum number queued messages. The
important point using bridges is their location. The nodes which communicate with each other frequently should be located in the same segment, since the performance improvement obtained will be strongly influenced by the message destination. As a result, bridged systems provide a better performance depending upon performance of the bridge, frame destination, local/remote message ratio, and location of the bridge in the system. Depending upon the application, the bridge interlinking gives a superior performance and the added cost consideration can be justified.

It may be concluded that the limitation on the maximum length of a single bus introduces trouble in CAN systems which are used in large areas. But bridges which are ideally located with respect to internode activity can solve problems in large CAN industrial applications.

7. References

[CANM95a] CAN is suitable for ex app., CAN newsletter, March 1995, pg. 12, Cia, Germany


[HALL96] Hallsov E., Data communication, computer networks, open systems, Addison-Wesley Pup. GM. 1996, USA.


[MCL95] McLauglin R., "CAN Controlling from cars to X-rays", IEE Networking, May 95, UK.


