ISSN: 2278 – 909X
International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE)
Volume 5, Issue 3, March 2016

Detection of Fault in CAN Bus and Diagnosis Algorithm

Saranya C
P.G Scholar,Embedded Systems,
BIT,
Sathyamangalam,India.

Karthick S
Associate Professor(ECE),
BIT,
Sathyamangalam,India.

***Abstract*-An Controller Area Network(CAN) bus is used in sending and receiving messages between devices in automobiles. During the transmission of messages through the nodes, there are possibilities of errors. To detect those errors an algorithm called Adaptive Fault Diagnosis algorithm for Controller Area Network(AFDCAN) is usedto detects all faulty nodes on the CAN.The algorithm uses single-channel communication expanding the bus-based classic standard CAN protocol like single path communication. Faults at nodes can arise due to failures in the system, in the memory. Bugged node or faulty node doesn't send or receive packets to neighbor nodes and send wrong packets to neighbor nodes. CANcentrate uses an active hub to connect the CAN based nodes and prevents the propagation of errors from one port to others. The individual test results are exchanged among these processors and the fault-free processors accuratelydiagnoses the actuator fault. Test rounds continue until the last node in the system is tested. The last fault-free node sends the second result frame to the earlier fault-free node after all of the test rounds are completed. One node can be tested multiple times by another node, and tests are conducted asynchronously. The number of test rounds required for a hierarchical adaptive distributed system level diagnosis algorithm Hi-ADSD is less than that for adaptive DSD. Parameters are fault detection time, bus load, no of faults occurred.**

***Keywords*-AFDCAN(Adaptive Fault Diagnosis algorithm for Controller Area Network), CAN Bus, Bugged nodes.**

I. INTRODUCTION

Network security has become more important tothe personal computers. The internet itself allowed for many security threats to occur. The basic architecture of the internet, when modified a little can reduce the possible attacks like hacking and bugs that are sent across the network. Already knowing the attack methods, allows for the appropriate security to emerge. Many businesses protect themselves from the internet by means of firewalls and encryption mechanisms. The businesses create a network to remain connected to the internet but secured from possible attacks.

Controller Area Network (CAN) bus used to find faults that is occurred in processing time. These faults need to be detected and diagnosed.Connection failures and node failures may lead to CAN network failure. Faults at nodes can arise due to failures in the controller, in the storage device, or in theinput–output peripherals. An algorithm for CAN that detects all faulty nodes on can adaptive fault -diagnosis algorithm for CAN bus is been proposed. It allows new node entry and re-entry of repairing faulty nodes during a diagnostic cycle. The robustness of CAN may be attributed in part to its abundant error-checking procedures.

The main objective of this paper is to

To detect fault nodes.
To reduce energy consumption.
To obtain single channel communication in CAN.

To detect the faults that are happening in the communication systems of CAN bus, detecting them and diagnosing them. Here, instead of removing the faulty parts a new reliable part is replaced for the faulty parts.

The performance of the CAN bus through its throughput is shown in the graphical form. The graph represents the performance of the CAN bus after replacing the faulty nodes with the reliable nodes. By detecting and checking the frequency match of the nodes some time is consumed and as the bus load increases the throughput is represented. As the number of nodes increases the number of packets that is lost is represented in graph.

II. DEVICES AND METHODS

1)Software Tools

Network simulator:

Front end is TCL
Back end is C++ event scheduler

NS is an object oriented discrete event simulator and an object oriented simulator, written in C++, with an Object oriented tool command language interpreter as a frontend. The root of this hierarchy

is the class TclObject. Users create new simulator objects through the interpreter and are reflected similarly by a corresponding object in the compiled hierarchy.

Advantages of Network Simulator:

Cheap, does not require costly equipment.

Scenarios that are complex can be easily tested.

Results can be quickly got and more ideas can be tested in a smaller time frame.

Controlled experimental conditions– Repeatability helps aid debugging.

Disadvantages:

Real systems are difficult to model

PROGRAMMING STRUCTURE

Build the event scheduler

Turn on tracing

Set up network topology

Set up transport connections

Generate traffic

Insert errors

III.EXPERIMENTAL SETUP AND PERFORMANCE

1)Diagnosis And Fault-Tolerant Control

Control systems appear in many products that are used in everyday life, but mostly remain unnoticed by their users. For example, control systems can be found in household appliances ranging from mobile phones to satellites.

But they can also be found in cars, ships, and aircraft. Under normal circumstances, control systems perform the tasks they are constructed for and therefore their users are unaware of them. When a fault occurs that prevents correct functioning of the system, this indeed gets noticed by the user.

2)Fault Detection

It is important to establish what events can be classified as a fault. A generally accepted definition of a fault is that it is anot permitted deviation that is varied in least one characteristic property or a parameter of a system from its acceptable/usual/standard condition. When an unpredicted change in the planned property of the system might result in different result and this is called as fault.

4)Error Capabilities

The CAN stipulation includes a Cyclic Redundancy Code to check errors on each frame's contents. Frames with errors are do not agree withall nodes, and an error frame can transmit error to the network system. The global and local faults are seperated by the controller, and if too many errors are detected, individual nodes can be stopped from transmitting errors or make it disconnect itself from the network thoroughly.

5)Modules:

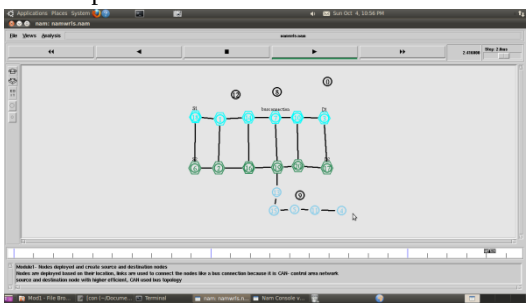
1.Nodes deployed and create source and destination node

2. Fault diagnosis in single channel communication

3. Rectifying faults in CAN

5.1)Nodes deployed and create source and destination node

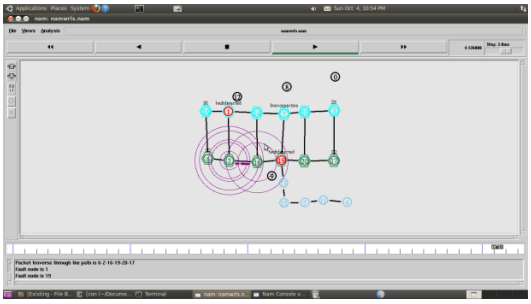
Nodes are deployed in the network with the help of NS2.



Nodesarranged and connected with controller

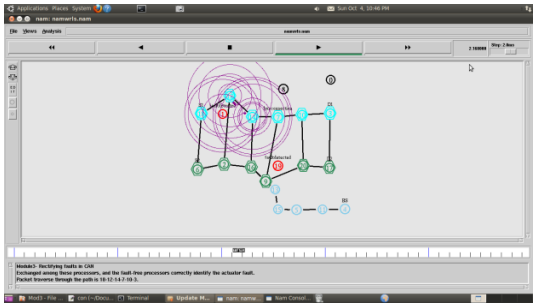
5.2)Fault diagnosis in single channel communication

Source node want to send packet to neighbor node connected in bus.



Detection of faulty nodes

5.3)Rectifying faults in CAN



checking newly connected nodes

6)CREATE NETWORK TOPOLOGY (PHYSICAL LAYER)

The Physical Layer is the first and the lowest layer in the OSI model of computer networking.

Cmn header
Ip header
Tcp header
Rtp header
Trace header

The Physical Layer consists of the essential hardware transmission technologies of a system network. It is a basic fundamental layer that is below all the logical data structures that belong in the higher level functions in a system network. The Physical Layer defines the means of transmitting raw bits rather than logical data packets over a physical link connecting networking chain nodes. The bit stream may be encoded into code words or symbols and converted to a physical that is transmitted over the network hardware.

8)Transport Connection (Transport Layer)

Transport layers are contained in both the TCP/IP protocols, which are the foundation of the network and the OSI model of general networking. The definitions of the Transport Layer are different in these two protocol models. This article primarily refers to the TCP/IP model, in which TCP is used largely for a convenient application programming interface to internet hosts, as opposed to the osi model of definition interface.

9)Generate Traffic (Application Layer)

In TCP/IP, the Application Layer contains all protocols and methods that fall into the realm of process-to-process communications via an Internet Protocol (IP) network using the Transport layer protocols to establish underlying host-to-host connections.

In the OSI model, the definition of its Application Layer is narrower in scope, explicitly distinguishing additional functionality above the Transport Layer at two additional levels: session layer and presentation layer OSI specifies strict modular separation of functionality of these layers and provides a protocol for each layer.

10)Feasibility Analysis

The feasibility analysis is actually done to check whether the problem is solvable or not. To check the feasibility so many analysis tests are done. But there are three main feasibility tests to be performed. They are

10.1)Operational Feasibility

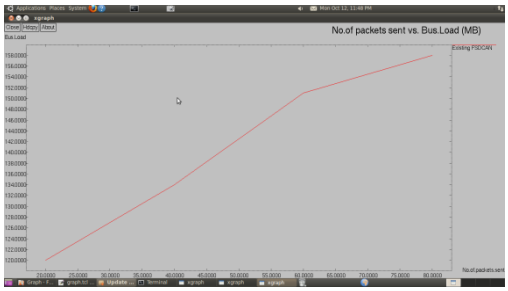
A thorough analysis is done and found that the system is functional and is operational. During analysis operational analysis is necessary. This analysis makes sure that this paper is operational.

10.2)Technical Feasibility

Actually the requirements for the application are very less and thus it is technically feasible. The technical feasibility checks the necessary products available for development of the paper.

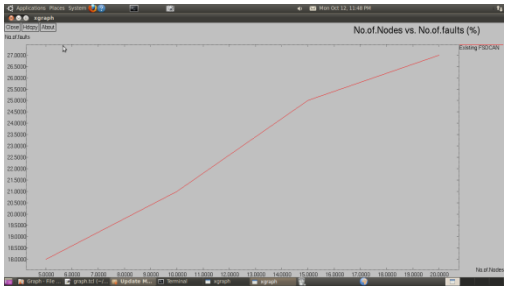
PERFORMANCE ANALYSIS

Performance analyzation:



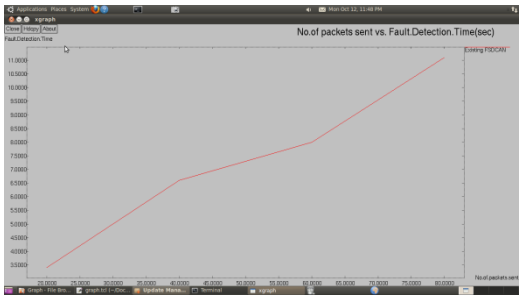
no. of packets sent vs no. of bus load

The Graph shows the amount of packets sent in according to the bus load.



no. of nodes vs no. of faults

The graph shows the number of packets lost for each number of nodes that is detected for fault.



no. of packets sent vs fault detection time

The time taken for the detection of fault during the number of packets sent is given in the graph.

Thus the graphs show the number of faults detected and the time taken to detect those faults. A disadvantage in the system is when the faulty nodes are replaced, to search for another node and again the packets are sent and verified weather the new node is a reliable node.

IV. CONCLUSION AND FUTURE WORK

CONCLUSION

The faults that are occurring in the CAN bus may be of bus load and some sort of bugs. These faults are detected by comparing each node with their IP address and their frequency. When compared, the faulty nodes are identified and the most near and reliable node is searched and is attached. Again the detection process happens and all the nodes are checked for faults, only if no error or bug is found the transmission of packet continues. The detection and correction of the faults delays in time to send the packets. The delay is occurred because to find a new node and also fault free node consumes some time to replace the faulty node with the new relay node. But the packet is been sent to the destination node correctly. The paper can be still developed with the deletion of the delay time that is taken to find new relay nodes, in replacement of the faults nodes.

The error is detected by transmission of packets. When the sent packets are not reached the destination then the fault is detected. In the simulation, the falling of packets is shown. And also the variation in parameters by throughput and the rate of faults in increase of bus load and number of nodes respectively is displayed in the graph.

FUTURE WORK

The detection and correction of the faults delays in time to send the packets. The delay is occurred because to find a new node and also fault free node consumes some time to replace the faulty node with the new relay node. But the packet is been sent to the destination node correctly. The paper can be still developed with the deletion of the delay time that is taken to find new relay nodes, in replacement of the faults nodes. By allocating the relay nodes previously the faults can be reduced.

The delay time can be reduced further by predicting the relay nodes in previous to replacing the faulty nodes. By reducing the time taken to check the nodes and by detecting it previously the time can be reduced.

V. REFERENCE

[1] SupriyaKelkar and Raj Kamal, “Adaptive Fault Diagnosis Algorithm for Controller Area Network” VOL. 61, NO.10, OCTOBER 2014.

[2] Cauffriez L, CiccotelliJ,Conrardc B, and Bayartc M, “Design of intelligent distributed control systems: A dependability point of view,” Reliab. Eng. Syst. Safety, vol. 84, pp. 19–32, 2004

[3] Fuhrer T, Muller B, Dieterie W, HartwichF,Hugel R, and Weiler R, “Time triggered communication on CAN,” in Proc. 7th Int. CAN Conf., Amsterdam, Netherlands, 2000.[Online].Available:http://www.Boschsemiconductors.de/media/pdf_1/canliteratur/cia2000paper_1.pdf

[4] Kelkar S and Kamal R, “Control area network based quotient remainder compression-algorithm for automotive applications,” in Proc. 38th Annu. IEEE IECON, Montreal, QC, Canada, Oct. 2012, pp. 3030–3036.

[5] Kelkar S and Kamal R, “Comparison and analysis of quotient remainder compression algorithms for automotives,” in Proc. IEEE INDICON, Kochi, India, Dec. 2012, pp. 802–807.

[6] Kopetz H and Grünsteidl G, “TTP-A protocol for fault-tolerant real-time systems,” Computer, vol. 27, no. 1, pp. 14–23, Jan. 1994.

[7] Manuel B, Julián P, Guillermo N, and Luís A, “An active star topology for improving fault confinement in CAN networks,” IEEE Trans. Ind. Informat., vol. 2, no. 2, pp. 78–85, May 2006

[8] Pimentel J R and Fonseca J A, “FlexCAN: A flexible architecture for highly dependable embedded applications,” in Proc. 3rd Int. Workshop Real-Time Netw., Italy, 2004. [Online]. Available: http://paws.kettering. edu//~jpimente/flexcan/FlexCAN-architecture.pdf

[9] Robert Bosch GmbH, Ver. 2.0 Controller Area Network (CAN)—Protocol Specification1991, Robert Bosch GmbH, Ver. 2.0.

[10] Short M J and Pont M J, “Fault-tolerant time-triggered communication using CAN,” IEEE Trans. Ind. Informat., vol. 3, no. 2, pp. 131–142, May 2007.