Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 4, April 2014, pg.597 – 606

RESEARCH ARTICLE

MONITORING OF RAILWAY BOGIES AND PREVENTING ACCIDENTS USING ARM7TDMI AND CAN PROTOCOLS

Siddu S Kadiwal, ICE Dept in M.I.T, Manipal University
(Email ID: siddguruk@gmail.com)
Dr. C. Gurudas Nayak, Associate Professor in ICE Dept in M.I.T, Manipal University
(Email ID: cgurudasnayak@yahoo.co.in)
Shobha Kadiwal, University B.D.T College
Gururaj Kadiwal, K.L.E'S College of Engineering & Technology

Abstract: This project proposes the new idea to continuously monitor the railway bogie ,Each bogie has electronic control unit which is connected sensors and all ECU are interconnected through can protocol bus ,here temperature is mainly monitored ,if fire accidents is occurred then emergency exits are opened and sprinkler will be on ,and accident is notified to main railway driver. Here the air condition in ac bogies is continuously monitored and controlled.

Introduction:

In these days train accidents are most common and the damage due to these accidents are more severe and takes many lives of passengers especially during night when all passengers are sleeping hence to reduce the accident rate due to fire accident we have to continuously monitor the all bogies all should be interconnected so we have to use can protocol for interconnection between the ECU.

As a serial communication protocol and defect in-vehicle network standard efficiently supporting distributed real-time control, the ranges of CAN (Controller Area Network) application domain is from high speed network to low cost multiplex wiring because of its special capabilities. In automobiles, electronic control units are connected using CAN.

1. ARM7TDMI

- The ARM7 processor is a member of the ARM family of general-purpose 32-bit microprocessors. The ARM family offers high performance for very low-power consumption and gate count.
- The ARM7TDMI-S is a general purpose 32-bit microprocessor, which offers high performance and very low power consumption. The ARM architecture is based on Reduced Instruction Set Computer (RISC) principles, and the instruction set and related decode mechanism are much simpler than those of micro programmed Complex instruction Set Computers. This simplicity results in a high instruction throughput and impressive real-time interrupt response from a small and cost-effective processor core.
- Pipeline techniques are employed so that all parts of the processing and memory systems can operate continuously. Typically, while one instruction is being executed, its successor is being decoded, and a third instruction is being fetched from memory.
- The ARM7TDMI-S processor also employs a unique architectural strategy known as THUMB, which makes it ideally suited to high-volume applications with memory restrictions, or applications where code density is an issue.
- The THUMB set‟s 16-bit instruction length allows it to approach twice the density of standard ARM code while retaining most of the ARM‟s performance advantage over a traditional 16-bit processor using 16-bit registers. This is possible because THUM code operates on the same 32-bit register set as ARM code.
- The key idea behind THUMB is that of a super-reduced instruction set. Essentially, the ARM7TDMI-S processor has two instruction sets.
- The standard 32-bit ARM instruction set.
- A 16-bit THUMB instruction set.

2. LPC2129 MICROCONTROLLER:

The LPC2129 is based on a 32 bit ARM7TDMI-S CPU with real-time emulation and embedded trace support, together with 256 kilobytes (kB) of embedded high speed flash memory. The architecture enables 32-bit code execution at maximum clock rate. For critical code size applications, the alternative 16-bit Thumb Mode reduces code by more than 30% with minimal performance penalty. With their compact 64 pin package, low power consumption, various 32-bit timers, combination of 4-channel 10-bit ADC and 2 advanced CAN channels and up to 9 external interrupt pins these microcontrollers are particularly suitable for industrial control, medical systems, access control. Number of available GPIOs goes up to 46 in 64 pin package.

2.1.Features:

• 32-bit ARM7TDMI microcontroller in a 64 pin package.
• 16 kB on-chip Static RAM.
• 256 kB on-chip Flash Program Memory (at least 10,000 erate/write cycles over the whole temperature range).
• Two interconnected CAN interfaces with advanced acceptance filters.
• Four channel 10-bit A/D converter with conversion time as low as 2.44 ms.
• Two 32-bit timers (with 4 capture and 4 compare channels), PWM unit (6 outputs), Real Time Clock and Watchdog.
• Multiple serial interfaces including two UARTs (16C550), Fast I2C (400 kbits/s) and two SPIs.
• 60 MHz maximum CPU clock available from programmable on-chip Phase-Locked Loop.
• Vectored Interrupt Controller with configurable priorities and vector addresses.
• Up to forty-six, 5 V tolerant general purpose I/O pins.
• On-chip crystal oscillator with an operating range of 1 MHz to 30 MHz
• Two low power modes Idle and Power-down.
• Processor wake-up from Power-down mode via external interrupt.

• Individual enable/disable of peripheral functions for power optimization.

• CPU operating voltage range of 1.65V to 1.95V (1.8V +/- 8.3%).

• I/O power supply range of 3.0V to 3.6V (3.3V +/- 10%)

3. HARDWARE DESIGN

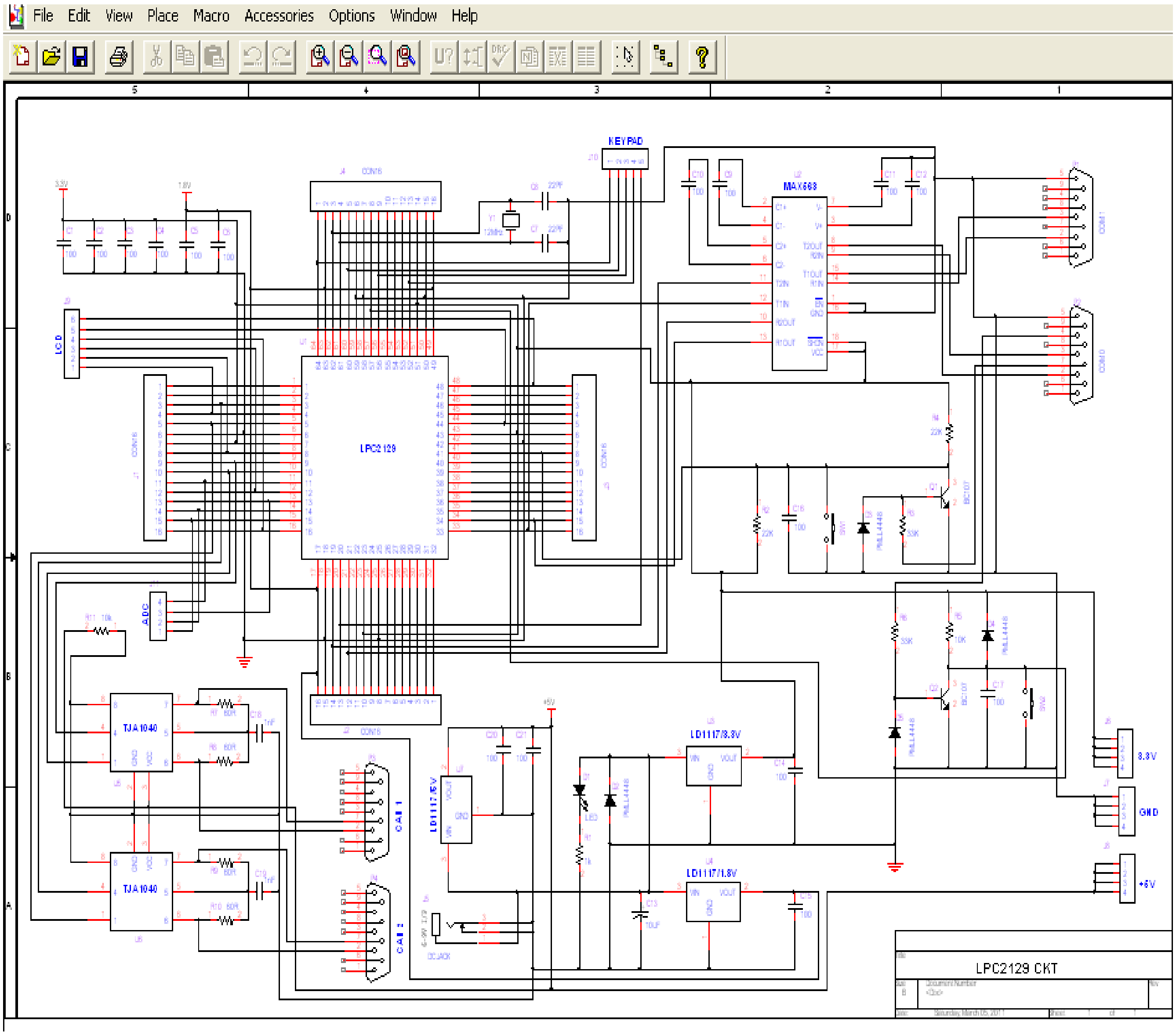


Fig . 3.1. Schematic Diagram of LPC 2129

4. CAN PROTOCOL

CAN or Controller Area Network is an advanced serial bus system that efficiently supports distributed control systems. It was initially developed for the use in motor vehicles by Robert Bosch GmbH, Germany, in the late 1980s, also holding the CAN license.

In the CAN protocol, the bus nodes do not have a specific address. Instead, the address information is contained in the identifiers of the transmitted messages, indicating the message content and the priority of the message. The number of nodes may be changed dynamically without disturbing the communication of the other nodes. Multicasting and Broadcasting is supported by CAN. CAN provide sophisticated error-detection and error handling mechanisms such as CRC check, and high immunity against electromagnetic interference. Erroneous messages are automatically retransmitted. Temporary errors are recovered. Permanent errors are followed by automatic switch-off of defective nodes. There is guaranteed system-wide data consistency.

The CAN protocol uses Non-Return-to-Zero or NRZ bit coding. For synchronization purposes, Bit Stuffing is used. There is a high data transfer rate of 1000 kilobits per second at a maximum bus length of 40 meters or 130 feet when using a twisted wire pair which is the most common bus medium used for CAN. Message length is short with a maximum of 8 data bytes per message and there is a low latency between transmission request and start of transmission. The bus access is handled via the advanced serial communications protocol Carrier Sense Multiple Access/Collision Detection with Non- Destructive Arbitration. This means that collision of messages is avoided by bitwise arbitration without loss of time.

4.1.CAN PROTOCOL VERSIONS

V2.0A (standard)-11 bit message ID

V2.0B (extended)-29 bit message ID

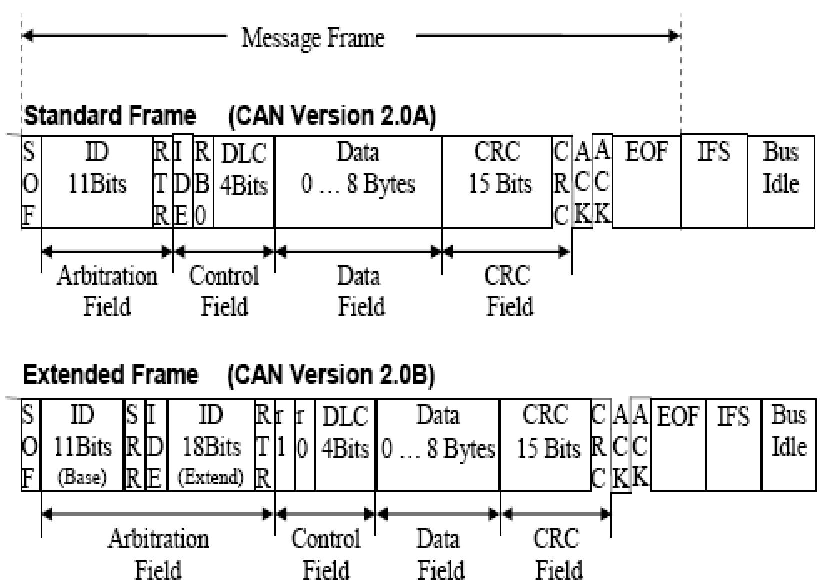


Fig. 4.1 CAN protocol versions

The original CAN specifications (Versions 1.0, 1.2 and 2.0A) specify an 11 bit message identifier. This is known as "Standard CAN". Those Data Frames and Remote Frames, which contain an 11-bit identifier, are therefore called Standard Frames. With these frames, 2^{11} (=2048) different messages can be identified (identifiers 0-2047).

4.2.CAN FRAMES

CAN having 5 types of frame formats

1. Data Frame

2. Remote Frame

3. Error Frame

4. Overload Frame

5. Interframe Space

4.3.ERROR DETECTION

The CAN protocol provides 5 sophisticated error detection mechanisms

1. CRC error

2. Acknowledge error

3. Form error

4. Bit error

5. Stuff error

4.4.FEATURES

• Supports 1 Mb/s operation
• Implements ISO-11898 standard physical layer requirements
• Suitable for 12V and 24V systems
• Detection of ground fault (permanent dominant) on TXD input
• Power-on reset and voltage brown-out protection
• An unpowered node or brown-out event will not disturb the CAN bus
• Low current standby operation
• Protection against damage due to short-circuit conditions
• Protection against high-voltage transients
• Automatic thermal shutdown protection

• Up to 112 nodes can be connected
• High noise immunity due to differential bus implementation
• Temperature ranges:
- Industrial (I): -40°C to +85°C
- Extended (E): -40°C to +125°C

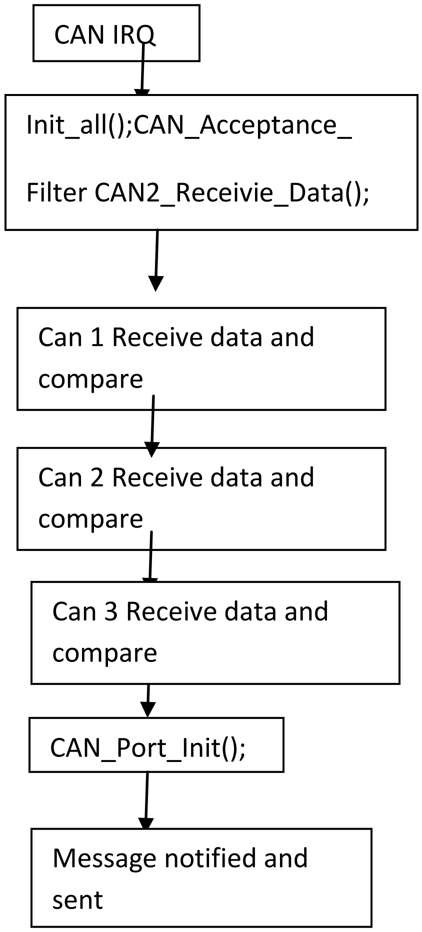
Pin Number	Pin Name	Pin Function
1	TXD	Transmit Data Input
2	VSS	Ground
3	VDD	Supply Voltage
4	RXD	Receive Data Output
5	VREF	Reference Output Voltage
6	CANL	CAN Low-Level Voltage I/O
7	CANH	CAN High-Level Voltage I/O
8	RS	Slope-Control Input

Table 4.2 CAN Transceiver

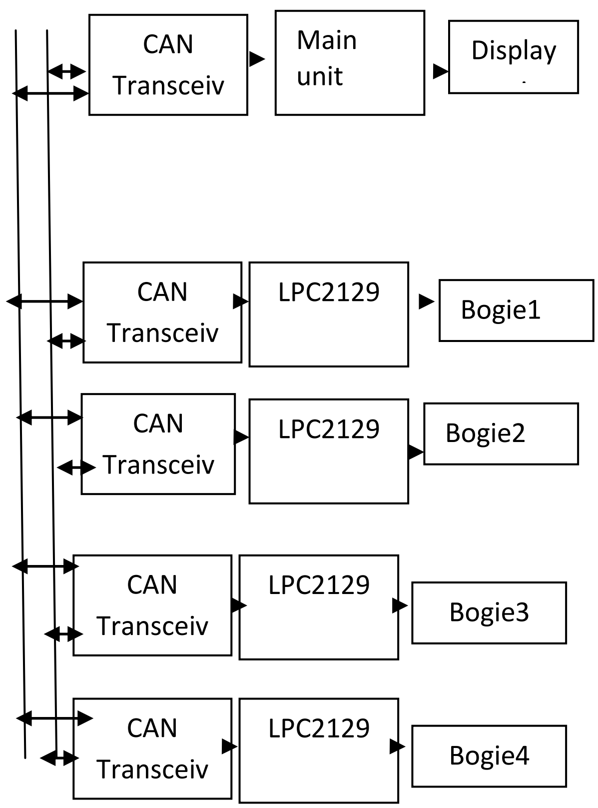
5. Working:

Each bogie has electronic control unit which is connected sensors mainly temp sensors that is LM35 which gives us Analog values again converted to digital values using ADC Channel, if temperature goes beyond certain threshold value then it is considered as fire accident so sprinkler going to be on, emergency exit will be open and alarm is also on. All Electronic Control Unit are interconnected through can protocol bus and continuously monitor the railway bogie and accident is notified to main railway driver. Here the air condition in ac bogie also continuously monitored using humidity sensor SY-HS-230 BT and controlled.

5.1.Flow Diagram:



5.2.BLOCK DIAGRAM:



Conclusions: In this project we are continuously monitor the different values of sensors, mainly temperature ,if fire accident occurred it is get notified and it will take necessary safety measures by itself automatically then we are able to reduce the damage and save many lives.
This is cost effective, by this we are able to increase safety features and comfort in Railway bogies.

ACKNOWLEDGEMENT
This Project would not have been possible without the support of my Guide. I like to express my gratitude and Indebtedness to my guide Prof. Dr. C. Gurudas Nayak.
Dr. C. Gurudas Nayak, received his. PhD from Manipal University. He has 22 years of Industrial/ Teaching experiences and published 40 papers in International, national Conferences and journals. He is currently working as an Professor(ICE) in M.I.T.Manipal University, Manipal. His research interests include of Communication Networks and Mobile Telephone Systems.
Mr Siddu S kadiwal, MTech(control systems) ICE dept in Manipal Institute of Technology, Manipal University Manipal.
Miss Shobha Kadiwal, Mtech(Power system and Power Electronics) University B.D.T College Davangere.
Mr.Gururaj Kadiwal, B.E(Computer Science)K.L.E'S College of Engineering & Technology, Belgaum.

REFERENCES

[1]. ARM Ltd., AMBA 4 AXI4, AXI4-Lite, and AXI4-Stream Protocol Assertions, 2011, www.arm.com

[2]. N. T. Slingerland and A. J. Smith., "Multimedia Instruction Sets for General Purpose Microprocessors: A Survey," University of California at Berkeley Computer Science, Technica

[3]. Mclauglin R..”CAN Controlling from cars to X-rays”,IEEENetworking May 95,UK.

[4]. “Controller Area Network (CAN),LAN in vehicle communication protocol“,SAE JI583 Mar90,SEA Information Report,pg.20.226-248

[5].Ekiz.H.Powner,ET,Kutlu.A,”Design and implementation of a CAN/CAN Bridge”,Proceedings of IEEE ISPAN“96 Conference,pg.507-503,Beijing,China,12-14 June 1996.

[6]. Automotive Electrical systems/CAN,Automotive Handbook,pg.777-778,3.Edition,Robert Bosch Gmbh.

[7] Kumar, M. A.Verma, and A. Srividya, Response-Time “Modeling of Controller Area Network (CAN). Distributed Computing and Networking,Lecture Notes in Computer Science Volume 5408, p 163-174, 2009.

[8] Tindell, K., A. Burns, and A.J. Wellings, Calculating controller area network (CAN) message response times. Control Engineering Practice, 3(8):p. 1163-1169, 2005.

[9] Li, M., Design of Embedded Remote Temperature Monitoring System based on Advanced RISC Machine. Electrotechnics Electric, 06, p. 273, 2009.

[10] Prodanov, W., M. Valle, and R. Buzas, A controller area network bus transceiver behavioral model for network design and simulation. IEEE Transactions on Industrial Electronics, 56(9): p. 3762-377, 2009.

[11] ISO (1993). Road Vehicles: Interchange of Digital Information: Controller Area Network (CAN) for High Speed Communication. ISO11898:1993.

[12] B.Gmbh, “CAN specification” vol 1 Version 2.0, 1991.

[13] Pazul, “Controller Area Network (CAN) Basics”, Microchip technology Inc., AN713, May 1999.

[14] Wilfried Voss, A comprehensive guide to controller area network, Copperhill Media Corporation, 2005-2008.

[15] Benjamin C Kuo, M. Farid Golnaraghi, Automatic Control systems, Eight edition, John wiley & sons., Inc 2003.

