ES623 Networked Embedded Systems

CAN Bus

02nd May 2013

CAN

- Controller Area Network
- CAN-bus is an ISO standard computer network protocol and bus standard, designed for microcontrollers and devices to communicate with each other without a host computer.
- Designed earlier for industrial networking but recently more adopted to automotive applications.
- CAN have gained widespread popularity for embedded control in the areas like industrial automation, automotives, mobile machines, medical, military and other harsh environment network applications.

History

- Development of the CAN-bus started originally in 1983 at Robert Bosch GmbH.
- Protocol was officially released in 1986.
- First CAN controller chips, produced by Intel and Philips, introduced in the market in the year of 1987
- Intel (82526) and Philips (82C200).

CAN Introduction

- The CAN is a "broadcast" type of bus.
- There is no explicit address in the messages.
- All the nodes in the network are able to pick-up or receive all transmissions.
- There is no way to send a message to just a specific node.
- Messages transmitted from any node on a CAN bus does not contain addresses of either the transmitting node, or of any intended receiving node.
- an identifier that is unique throughout the network is used to label the content of the message.

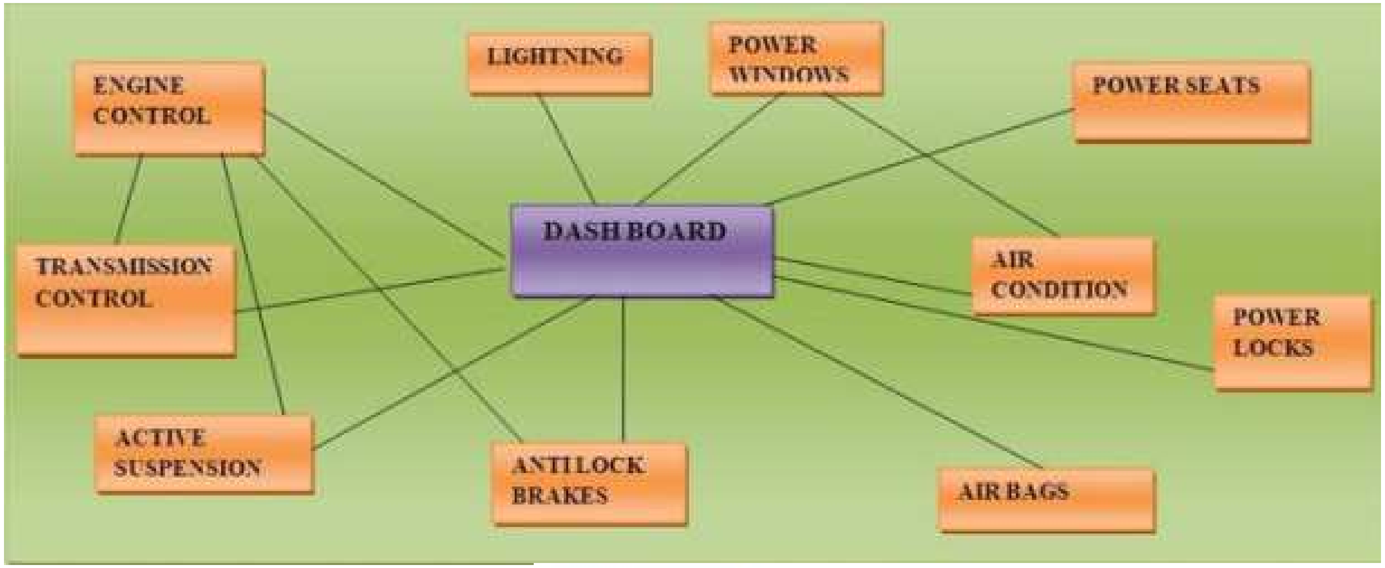
CAN Introduction

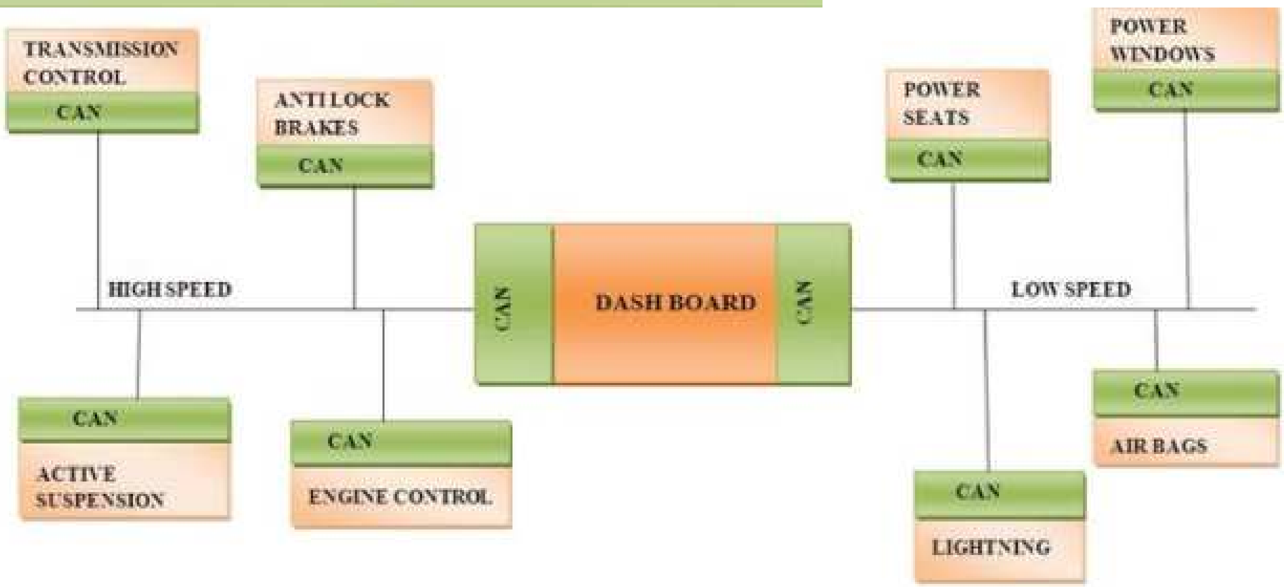
- Each message carries a numeric value, which controls its priority on the bus, and may also serve as an identification of the contents of the message.

- And each of the receiving nodes performs an acceptance test or provides local filtering on the identifier to determine whether the message, and thus its content, is relevant to that particular node or not, so that each node may react only on the intended messages.

- If the message is relevant, it will be processed; otherwise it is ignored.

Automobile application





How CAN Communicate

- In situations where two or more nodes attempt to transmit message (to the CAN bus) at the same time, the identifier field, which is unique throughout the network helps to determine the priority of the message.
- A "non-destructive arbitration technique" is used to accomplish this, to ensure that the messages are sent in order of priority and that no messages are lost.
- The lower the numerical value of the identifier, the higher the priority.

How CAN Communicate

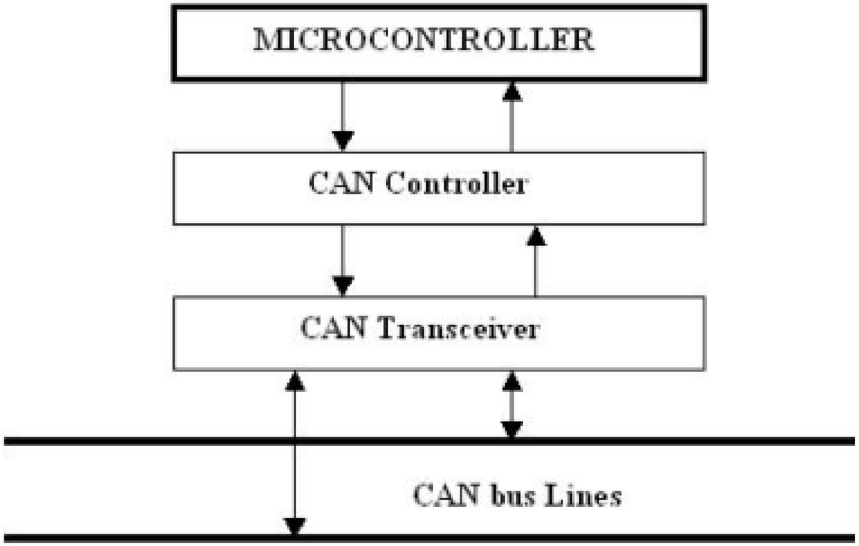
- CAN do not use address-based format for communication, instead uses a message-based data format.

- Here the information is transferred from one location to another by sending a group of bytes at one time (depending on the order of priority).

- This makes CAN ideally suited in applications requiring a large number of short messages (e.g.: transmission of temperature and rpm information). by more than one location and system-wide data consistency is mandatory.

How CAN Communicate

- CAN standard facilitates the subsystem to control actuators or receive signals from sensors.
- A CAN message never reaches these devices directly, but instead a host-processor and a CAN Controller (with a CAN transciever) is needed between these devices and the bus.



How CAN Communicate

- CAN Controller stores received bits (one by one) from the bus until an entire message block is available
- Then be fetched by the host processor (usually after the CAN Controller has triggered an interrupt).
- The CAN transciever adapts signal levels from the bus, to levels that the CAN Controller expects and also provides a protective circuitry for the CAN Controller.
- The host-processor decides what the received messages mean, and which messages it wants to transmit itself.

How CAN Communicate

- The more rapidly changing parameters need to be transmitted more frequently and, therefore, must be given a higher priority
- the priority of a CAN message is determined by the numerical value of its identifier
- When one node transmits the message, sometimes many nodes may accept the message and act on it
- For example, a temperature-sensing node may send out temperature data that are accepted & acted on only by a temperature display node.
- But if the temperature sensor detects an over-temperature situation, then many nodes might act on the information.

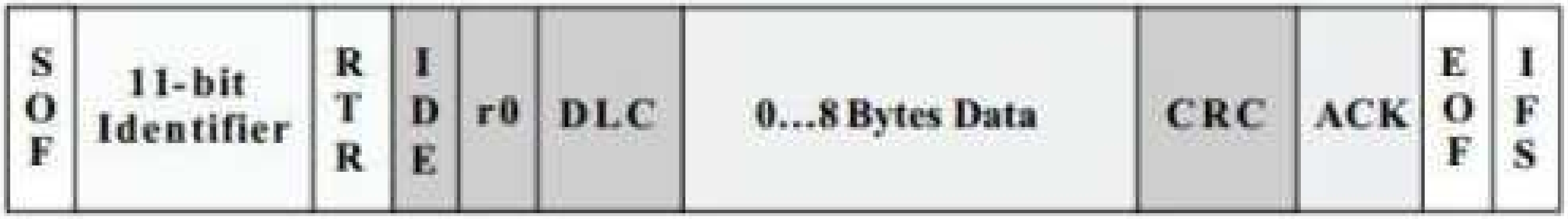
Features

- CAN use "Non Return to Zero" (NRZ) encoding (with "bit-stuffing") for data communication on a "differential two wire bus".
- CAN bus offers a high-speed communication rate up to 1 M bits / sec, for up to 40 feet, thus facilitating real-time control.
- It's possible to add nodes to the bus without reprogramming the other nodes to recognize the addition or changing the existing hardware.
- This can be done even while the system is in operation. The new node will start receiving messages from the network immediately. This is called "hot-plugging"

Features

- CAN protocol has the ability of a node to request information from other nodes. This is called a remote transmit request, or RTR.
- CAN protocol can link up to 2032 devices (assuming one node with one identifier) on a single network. But accounting to the practical limitations of the hardware (transceivers), it may only link up to 110 nodes on a single network.
- Has an extensive and unique error checking mechanisms.
- Has High immunity to Electromagnetic Interference.
- Has the ability to self-diagnose & repair data errors.

CAN Frames



SOF	11-bit Identifier	RTR	IDE	r0	DLC	0…8 Bytes Data	CRC	ACK	EOF	IFS

- SOF - Start of Frame bit.
- IDENTIFIER - It serves dual purpose one, to determine which node has access to the bus and second to identify the type of message.
- RTR - Remote Transmission Request. It identifies whether it's a data frame or a remote frame.

CAN Frames

- DLC – Data Length Code. It is 4 bit data length code that contains the number of bytes being transmitted.
- DATA– Used to store up to 64 data bits of application data to be transmitted.
- CRC– Cyclic Redundancy Check. The 16-bit (15 bits plus delimiter) cyclic redundancy check (CRC) contains the checksum of the preceding application data for error detection.
- ACK – Acknowledge (ACK) field. It compromises of the ACK slot and the ACK delimiter. When the data is received correctly the recessive bit in ACK slot is overwritten as dominant bit by the receiver.
- EOF– End of Frame (EOF). The 7-bit field marks the end of a CAN frame (message) and disables