UNIT II
EMBEDDED NETWORKING

2.6 SERIAL BUS COMMUNICATION PROTOCOLS–I2C

Inter connecting number of device circuits, Assume flash memory, touch screen, ICs for measuring temperatures and ICs for measuring pressures at a number of processes in a plant.

_ICs mutually network through a common synchronous serial busI2C An Integrated Circuit'(I2C) bus a popular bus for these circuits.

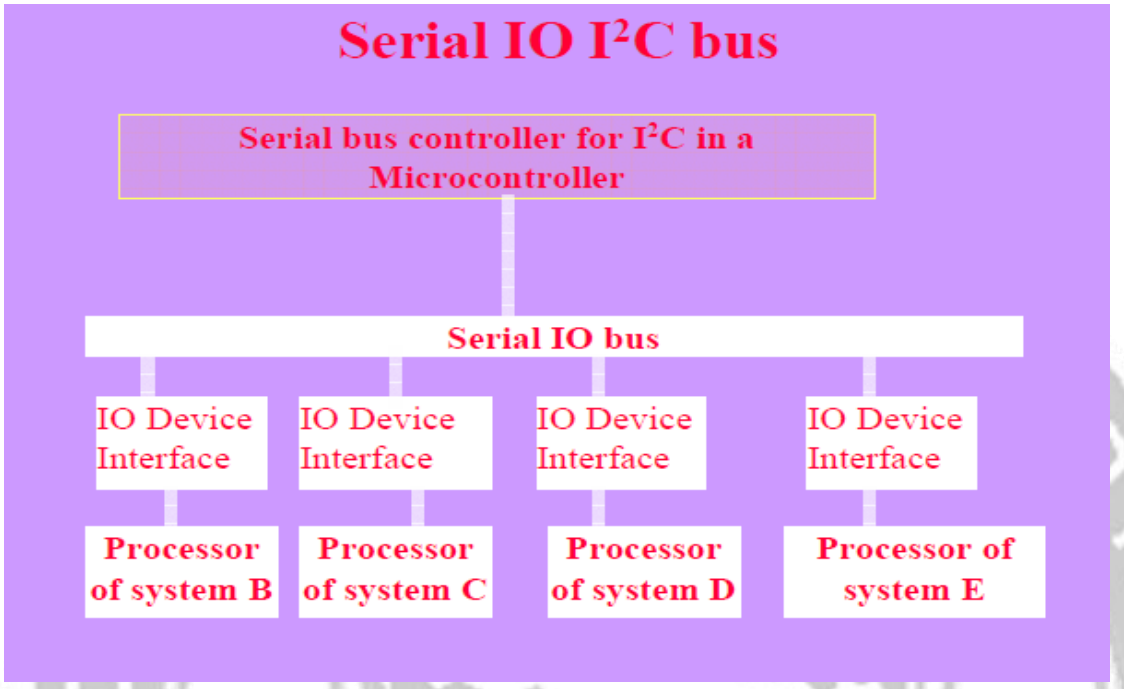
_Synchronous Serial Bus Communication for networking

_ Each specific I/O synchronous serial device may be connected to other using specific interfaces , for example, with I/O device using I2C controller

_ I2C Bus communication− use of only simplifies the number of connections and provides a common way (protocol) of connecting different or same type ofI/O devices using synchronous serial communication

IO I2C Bus

_ Any device that is compatible with a I2Cbus can be added to the system (assuming an appropriate device driver program is available), and a I2C device can be integrated into any system that uses that I2Cbus.



Originally developed at Philips Semiconductors

Synchronous Serial Communication 400kbps up to 2m and 100 kbps for longer distances

Three I2C standards

1. Industrial100kbpsI2C,
2. 100kbpsSMI2C,
3. 400kbps I2C

I2C Bus

_ The Bus has two lines that carry its signals— one line is for the clock and one is for bi-directional data.

_There is a standard protocol for the I2C bus.

Device Addresses and Master in the I2C bus

_Each device has a 7-bit address using which the data transfers take place.

_Master can address 127 other slaves at an instance.

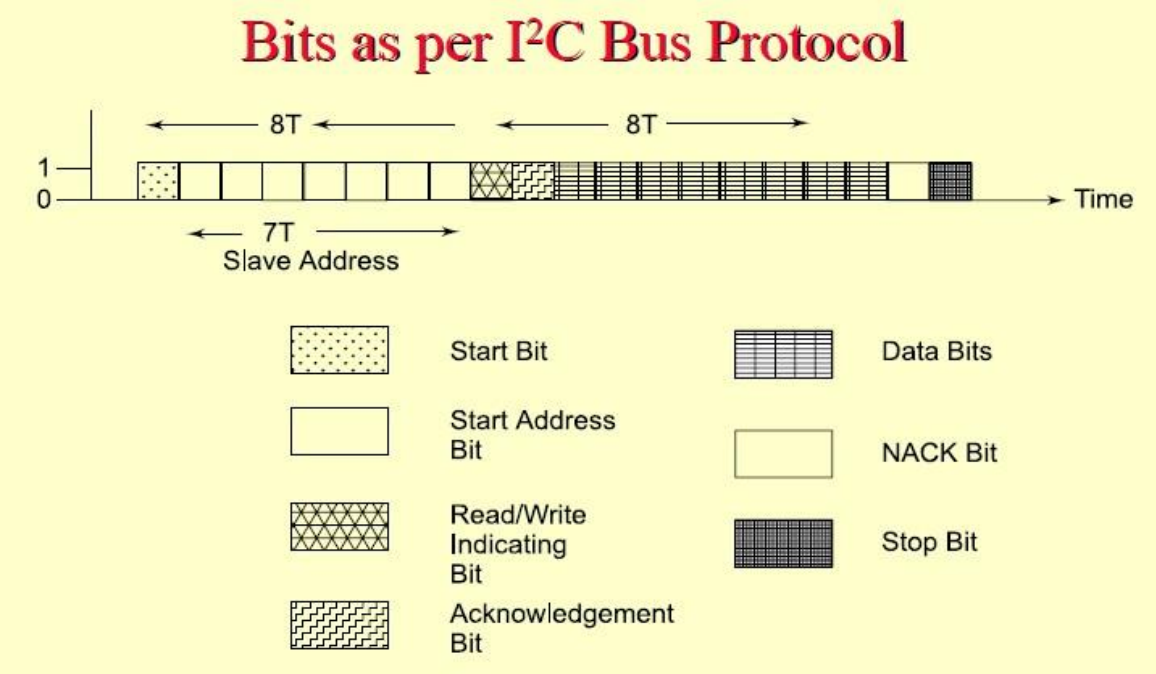
_ Master has at a processing element functioning as bus controller or a microcontroller with I2C (Inter Integrated Circuit)bus interface circuit.

Slaves and Masters in the I2C bus

_ Each slave can also optionally has I2C (InterIntegrated Circuit) bus controller and processing element.

_Number of masters can be connected on the bus.

_ However, at an instance, master is one which initiates a data transfer on SDA (serial data) line and which transmits the SCL (serial clock) pulses .From *master* a data frame has fields beginning from start bit



Synchronous Serial Bus Fields and its length

_First field of1 bit ─ Start bit similar to one in an UART

_ Second field of 7 bits─ address field. It defines the slave address, which is being sent the data frame (of many bytes) by the master

_Third field of 1 control bit─ defines whether a read or write cycle is in progress

_ Fourth fieldof1control bit─ defines whether is the present data is an acknowledgment (from slave)

_Fifth field of 8 bits─ **I2C device data byte**

_Sixth field of 1-bit─ bit NACK (negative acknowledgement) from the receiver. If active then acknowledgment after a transfer is not needed from the slave, else acknowledgement is expected from the slave

_Seventh field of1 bit ─ stop bit like in an UART

Disadvantage of 12C bus

• Time taken by algorithm in the hardware that analyzes the bits through I2C in case the slave Hardware does not provide for the hardware that supports it.

• Certain IC support the protocol and certain do not.

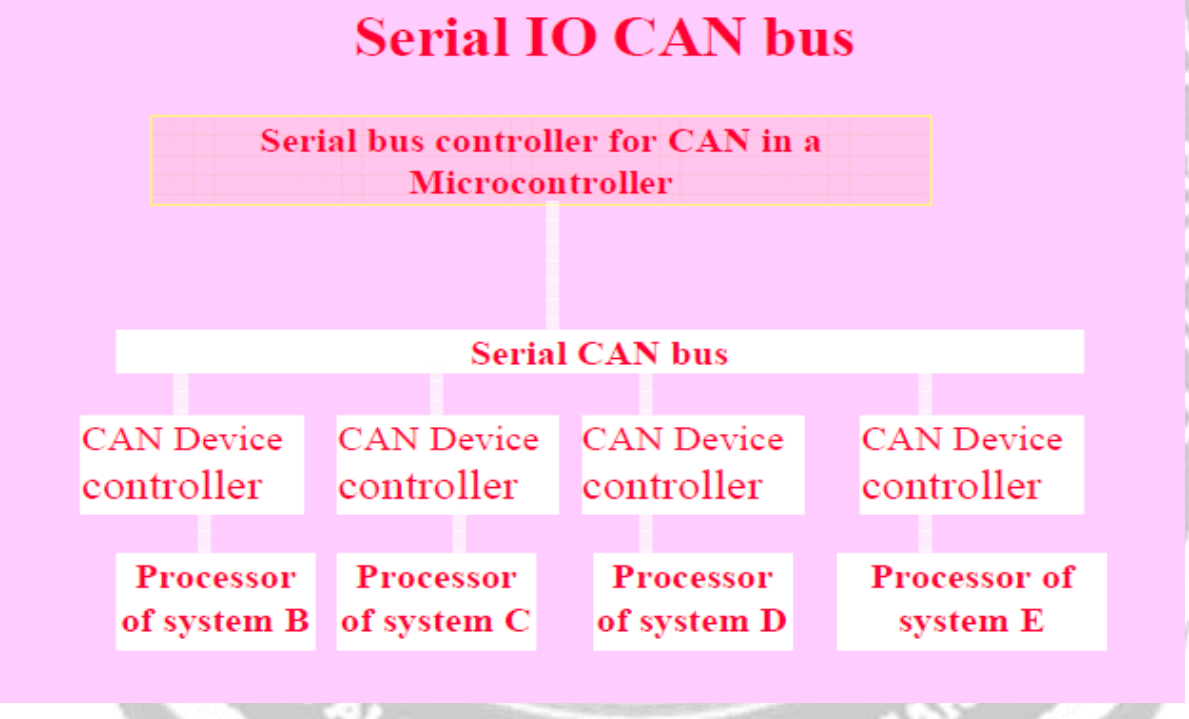
• Open collector drivers at the master need a pull-up resistance of 2.2 K on each line

SERIAL BUS COMMUNICATION PROTOCOLS – CAN

Distributed Control Area Network example- a network of embedded systems in automobile

_CAN-bus line usually interconnects to a CAN controller between line and host at the node. It gives the input and gets output between the physical and data link layers at the host node.

_ The CAN controller has a BIU (bus interface unit consisting of buffer and driver), protocol controller, status – cum control registers receiver-buffer and message objects. These units connect the host node through the host interface circuit



Three standards:

1. 33kbps CAN,
2. 110kbpsFault Tolerant CAN,
3. 1Mbps High Speed CAN

CAN protocol

There is a CAN controller between the CAN line and the host node.

_ CAN controller ─ BIU(Bus Interface Unit) consisting of a buffer and driver

_ Method for arbitration─ CSMA/AMP (Carrier Sense Multiple Access with Arbitration on Message Priority basis)

Each Distributed Node Uses:

• Twisted Pair Connection up to 40m–forbi-directional data

• Line which pulls to Logic 1 through a resistor between the line and +4.5Vto+12V.

• Line Idle state Logic1(Recessive state)

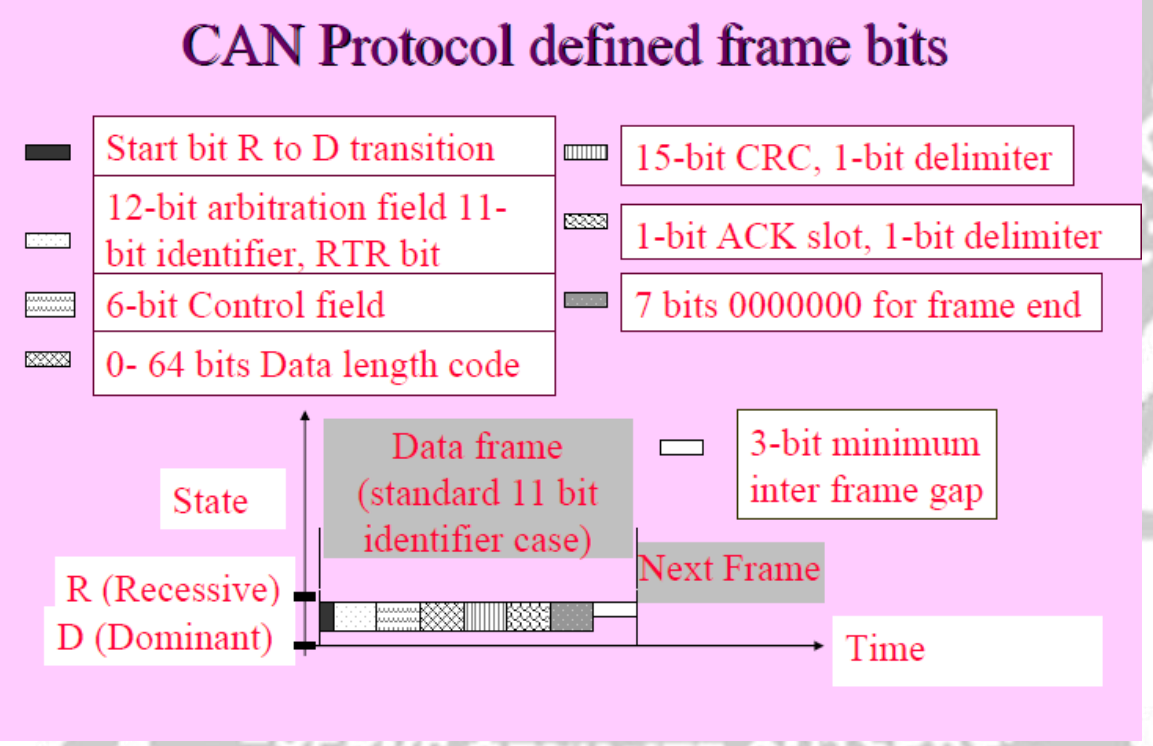
• Uses a buffer gate between an input pin and the CAN line

• Detects Input Presence at the CAN line pulled down to dominant (active) state logic 0 (ground ~0V) by a sender to the CAN line

• Uses a current driver between the output pin and CAN line and pulls line down to dominant (active) state logic 0(ground ~ 0V) when sending to the CAN line Protocol defined start bit followed by six fields of frame bits Data frame starts after first detecting that dominant state is not present at the CAN line with logic 1(R state)to 0(D state transition) for one serial bit interval

• After start bit, six fields starting from arbitration field and ends with seven logic0send-field

• 3-bit minimum inter frame gap before next start bit (R→ D transition)occurs



Protocol defined First field in frame bits

_First field of 12 bits ─ arbitration field

_ 11-bit destination address and RTR bit

(Remote Transmission Request)

_ Destination device address specified inan11-bitsub-field and whether the data byte being sent is a data for the device or a request to the device in 1-bitsub-field.

_ Maximum 211devices can connect a CAN controller in case of 11-bit address field standard11-bit address standard CAN

_Identifies the device to which data is being sent or request is being made.

_When RTR bit is at '1', it means this packet is for the device at destination address. If this bit is at'0'(dominant state) it means, this packet is a request for the data from the device.

Protocol defined frame bits Second field

_Second field of 6 bits─ control field.

The first bit is for the identifier ‘sex tension.

_The second bit is always '1'.

_ The last 4bits specify code for data Length

_Third field of 0to 64 bits ─Its length depends on the data length code in the control field.

• Fourth field (third if data field has no bit present) of 16 bits─ CRC (Cyclic Redundancy Check) bits.

• The receiver node uses it to detect the errors , if any, during the transmission

• Fifth field of 2 bits ─ First bit 'ACK slot'

• ACK='1'and receiver sends back'0' in this slot when the receiver detects an error in the reception .

• Sender after sensing'0' in the ACK slot generally retransmits the data frame.

• Second bit 'ACK delimiter' bit. It signals the end of ACK field.

• If the transmitting node does not receive any acknowledgement of data frame within a Specified time slot it should retransmit.

Sixth field of 7-bits─end-of- the frame specification and has seven'0's