# CANbus: A brief introduction

Incorporating: The Fujitsu status

Dave Mills

Queen Mary, University of London

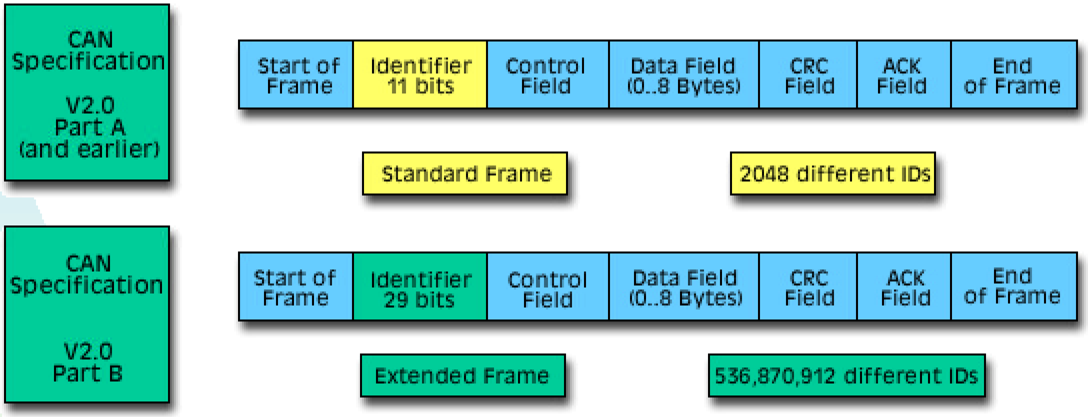
Queen Mary
University of London

# What is CAN?

- CAN stands for Controller Area Network
- CAN is a multimaster serial network.
- CAN defines the hardware level upon which higher level protocols can work e.g. CANOpen.
- The network is priority based, so that a high priority message will not be held up by a low priority message.(bus arbitration)

Queen Mary
University of London

# A standard CAN frame



CAN Specification

V2.0
Part A
(and earlier)

| Start of Frame | Identifier 11 bits | Control Field | Data Field (0..8 Bytes) | CRC Field | ACK Field | End of Frame |
|---|---|---|---|---|---|---|

Standard Frame

2048 different IDs

CAN Specification

V2.0
Part B

| Start of Frame | Identifier 29 bits | Control Field | Data Field (0..8 Bytes) | CRC Field | ACK Field | End of Frame |
|---|---|---|---|---|---|---|

Extended Frame

536,870,912 different IDs

There are two standards for CAN frames V2a and V2b. In 2a there are 11 bits set aside for identification, in 2b there are 29 bits available.
Both standards allow for a maximum of 8 bytes of data to be transmitted per message.

Queen Mary
University of London

# CAN Identifiers

Each node in a CAN network has a unique 11 bit (or 29 bit) ID. The ID is used to decide the priority of the message.

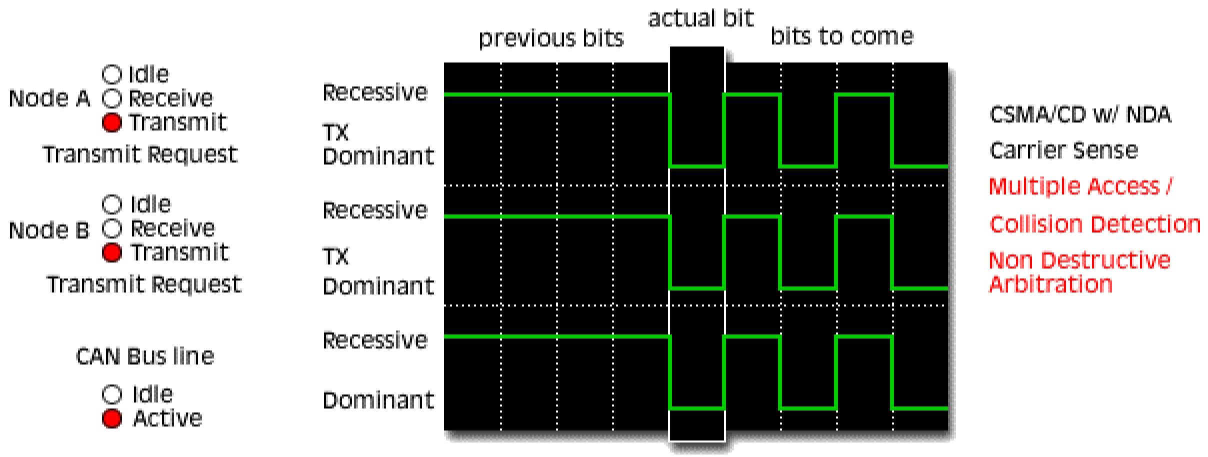
Messages with low values (00000000000 as opposed to 11111111111) are the highest priority.

CANOpen claims the first 4 bits of the ID to assign the message object type, this leaves us with 7 bits to identify the modules in a crate with.

Queen Mary
University of London

# How bus arbitration works

- If after finding the bus idle, two or more bus nodes start their transmission simultaneously, collision of the messages is avoided by bit-by-bit arbitration
- Each node sends the bits of its message identifier and monitors the bus level. As long as the identifier bits are identical, each node keeps transmitting its message
- The animation shows both node A and B transmitting until node B sends high and receives low, node B then switches to receiving while node A continues to transmit



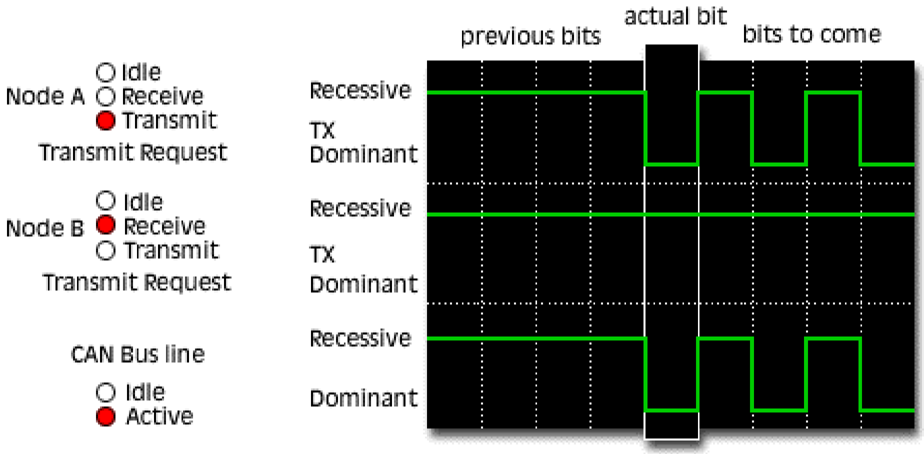
Queen Mary
University of London

# Message acknowledgement

| | | | | | | |
|---|---|---|---|---|---|---|
| Start of Frame | ID 11 Bits | Control Field | Data Field (0..8 Bytes) | CRC Field | ACK Field | End of Frame |

The highlighted field is the Acknowledge Field. The Acknowledge Slot bit is transmitted as a recessive bit. Any node that has received an error free frame acknowledges the correct reception of the frame by sending back a dominant bit, regardless of whether the node is configured to accept that specific message or not.

- •The transmitter checks to see if the ACK field is dominant
- •If so then at least 1 other node has Rx’ed the message correctly
- •If not then the message is sent again. An internal error counter is also incremented.

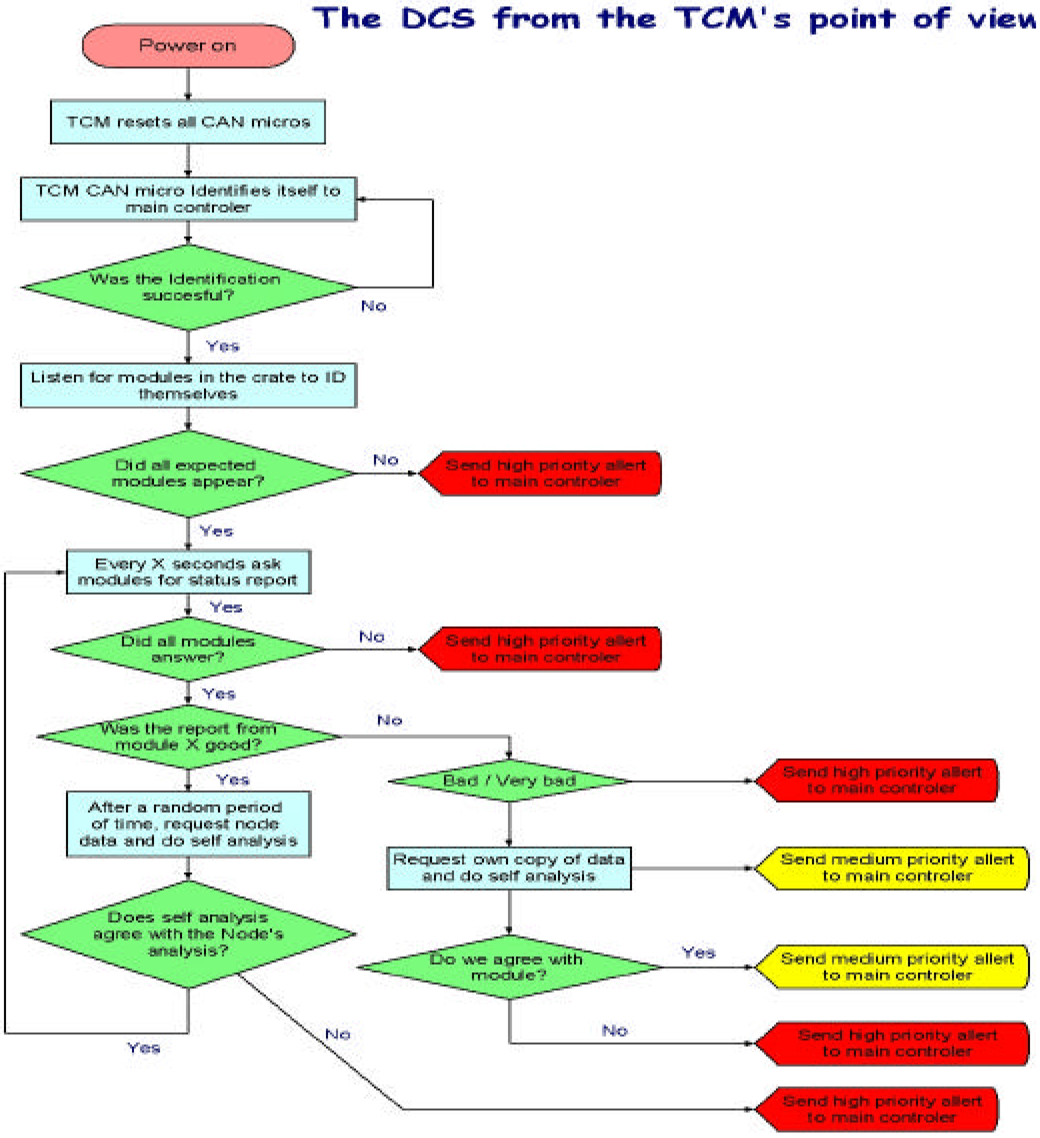


Queen Mary
University of London

# System Function 1

A possible system function path. This is the system that would be running on top of the CANOpen protocol.

This assumes that the TCM has some knowledge of what is in the crate and handles all communication with the outside world.



The DCS from the TCM's point of view

Power on

TCM resets all CAN micros

TCM CAN micro Identifies itself to main controler

Was the Identification succesful?

No

Yes

Listen for modules in the crate to ID themselves

Did all expected modules appear?

No

Send high priority allert to main controler

Yes

Every X seconds ask modules for status report

Yes

Did all modules answer?

No

Send high priority allert to main controler

Yes

Was the report from module X good?

No

Bad / Very bad

Send high priority allert to main controler

Request own copy of data and do self analysis

Send medium priority allert to main controler

Do we agree with module?

Yes

Send medium priority allert to main controler

No

Send high priority allert to main controler

Yes

After a random period of time, request node data and do self analysis

Does self analysis agree with the Node's analysis?

Yes

No

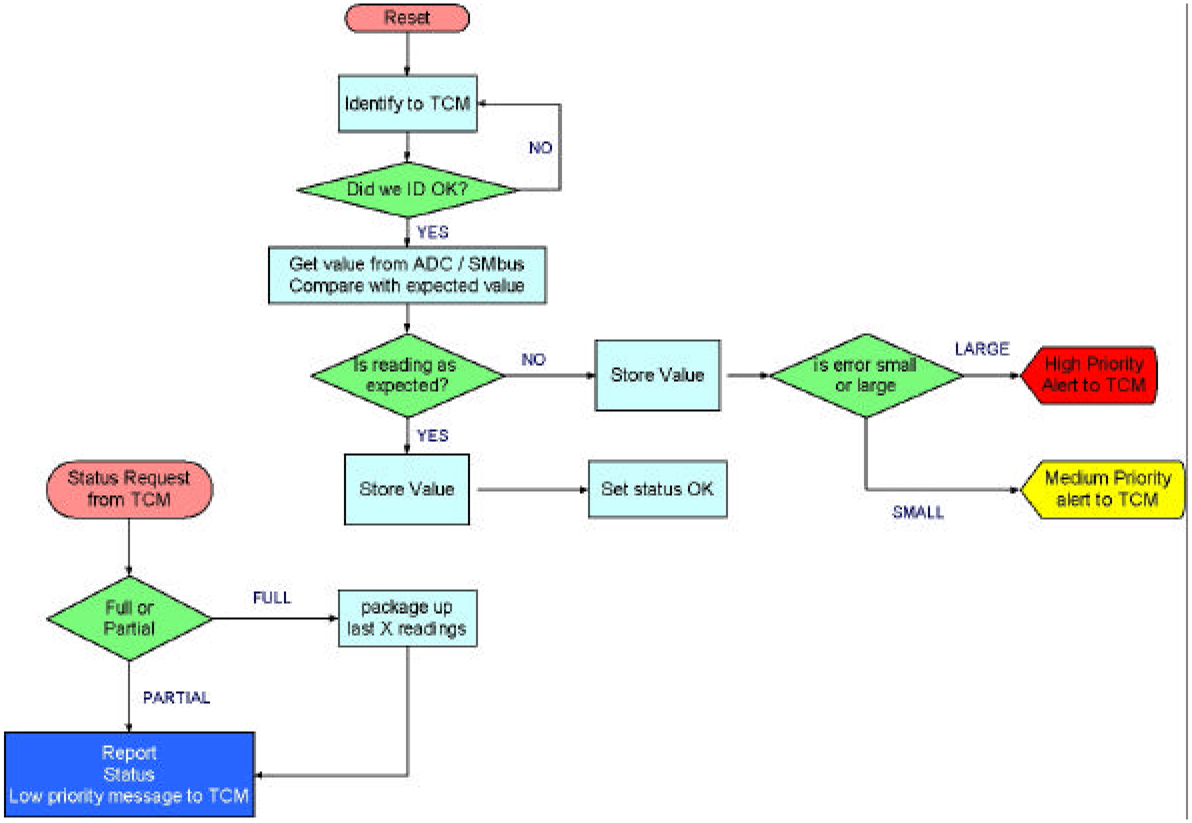
Send high priority allert to main controler

Queen Mary
University of London

# System Function 2

Module to TCM

This shows the code that may run on each node in the crate. All expected values of voltages and temperatures are programmed into the node, readings are stored in RAM until requested or overwritten with fresh data



# Fujitsu Status

- Most of the code is now tidy and readable
- Now documenting the code and the API to the code
- Need to specify the message data format
- Need to specify exactly how the network will operate
- Need to get a demo system running to test the code

Queen Mary
University of London