

# CANbus Status

---

Progress on the Fujitsu CANbus controller.

Adam Davis & David Mills



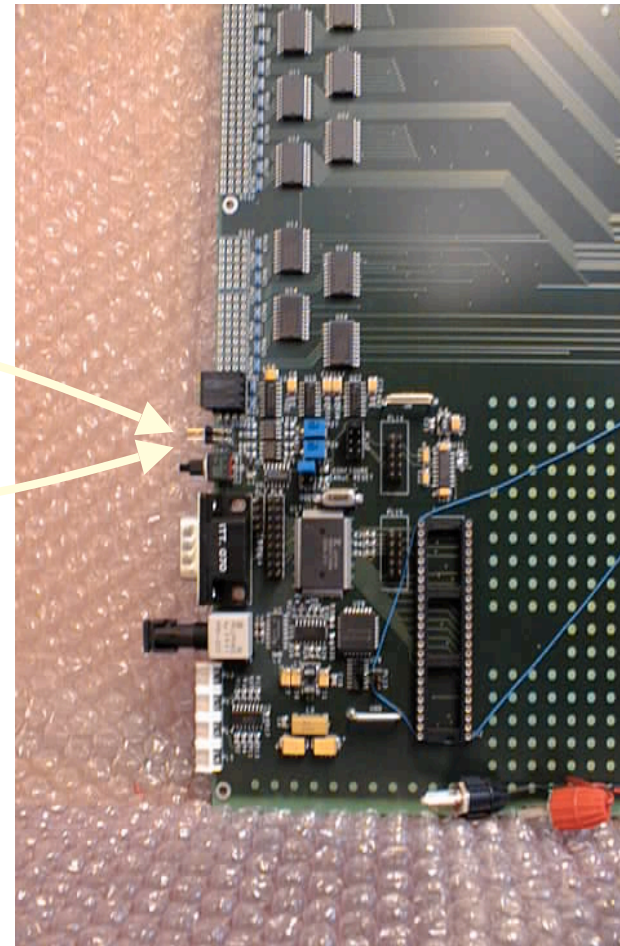
# Quick overview

- The Fujitsu is looking like a usable device now.
- CAN functionality has been tested in a variety of small systems.
- A library of modules has been written.
- A simplified CANOpen control program has been tested and works.



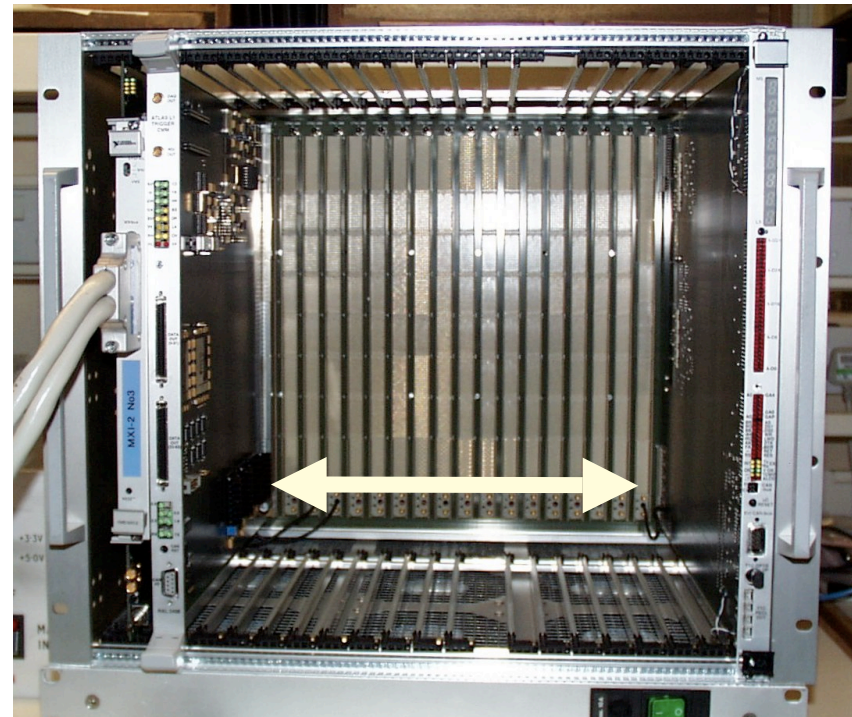
# Basic CAN communication

- A simple program was written to count CAN frames and display data on LEDs

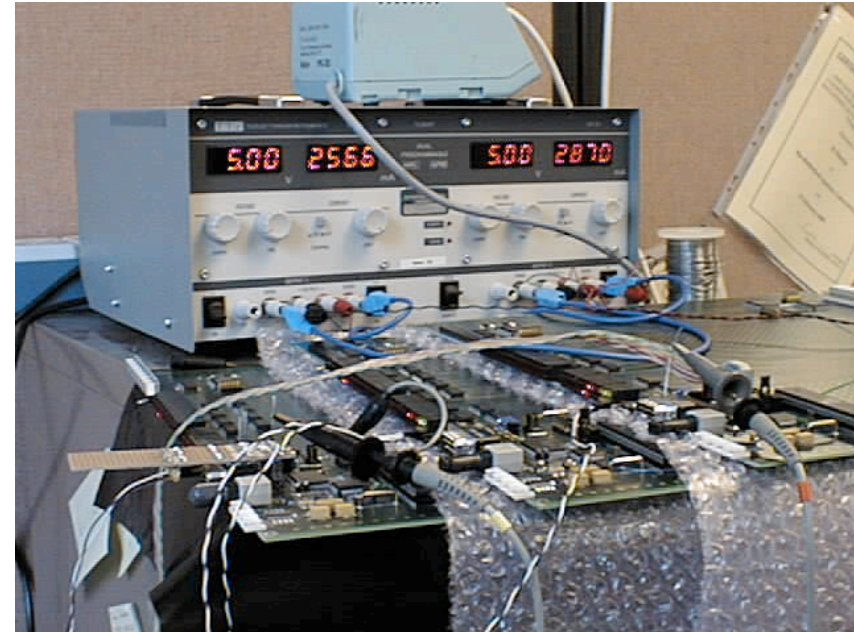
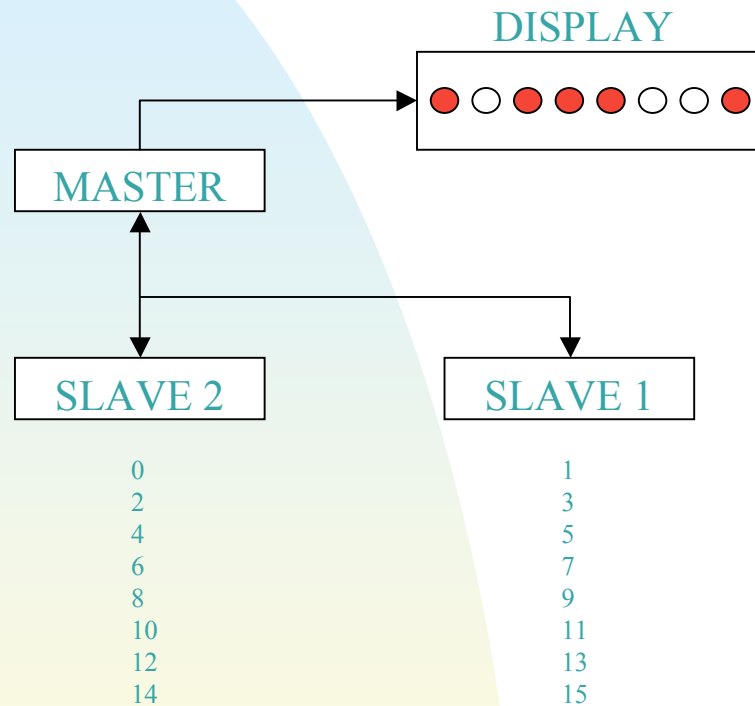


# CMM > TCM communication

- Established CAN communication between CMM and TCM in processor crate.
- Transfers data port from CMM to TCM.



# Simple CAN network

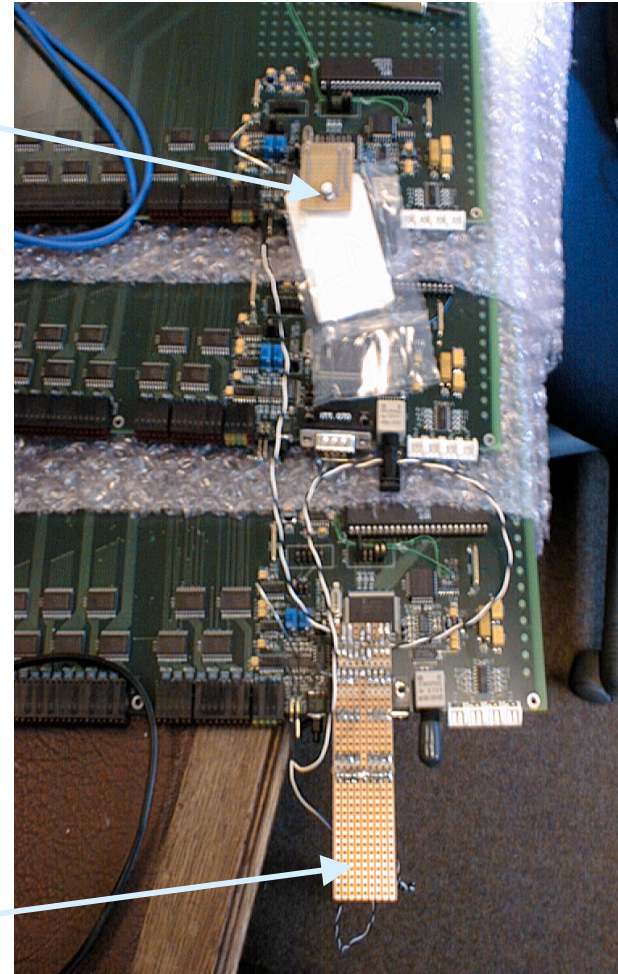


- Network using Three TCMs
- Utilises remote request function of CAN protocol
- Uses Reload Timer Interrupt for delay

# Sending real data over CAN

- Remote Request ADC Read via CAN Bus

ADC Port



LED Display

# Control Program

- A simplified version of the control program (CANOpen layer) has been tested on my HAL.
- The program sent out CANOpen node status messages and boot up messages.
- The next step is to integrate the other modules that I've written with this code.



# The Control Program test hardware

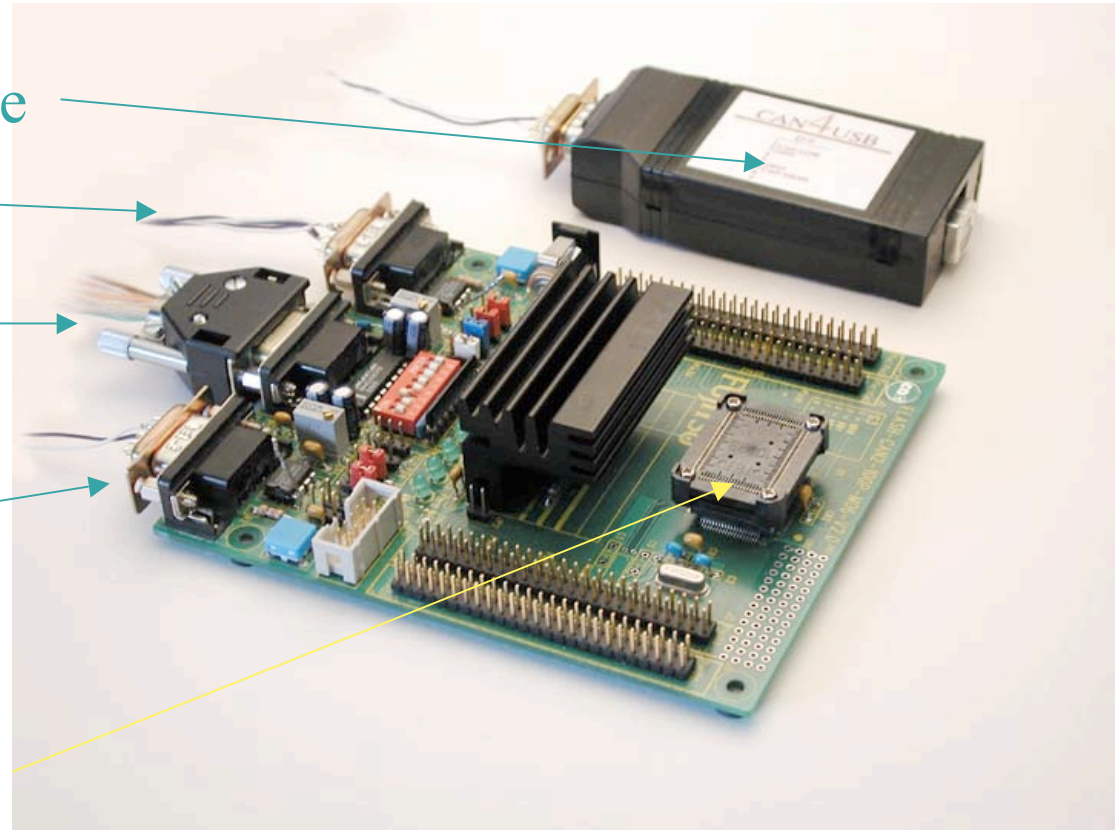
CAN > USB Interface

CAN0

RS232

CAN1

Micro-controller





# Software status

## ■ Adam:

- CAN access code written
- ADC & Interrupt code written
- Smbus code in development
- GEOADDR code in development

## ■ Dave:

- HAL has been tidied up and tested (ADC, DPM, VME, CAN, Timers).
- Control Program is still in development.
- A new Windows 'front end' is in development for the CAN > USB interface.



# To do

## ■ Adam

- Develop system with above “modules”:
  - ◆ CMM / CPM respond to TCM remote request
    - ✦ ADC Read (monitoring voltages)
    - ✦ FPGA Temperature
  - ◆ CMM / CPM send emergency message to TCM
    - ✦ Voltage above set threshold
    - ✦ Temperature above set threshold

## ■ Dave

- Add more functionality to the Control Program
- Add SMBus functionality to the HAL
- Test the system on real hardware.

