

**Design Journal**

**Project : DAQ Module (CAN)**

**Description :**

Firmware for

CAN logging and wireless interface for data acquisition

**Author(s) :**

Raphael Neveu

**Design**

**2016/07/24**

The DAQ module (data acquisition and logging) will have the main following features :

* Logging CAN trace to SD card
* Sending over Wifi the received CAN messages
* Transmit CAN messages sent from a PC (lower priority)

We could hard code the configuration, but it is smarter to make an flexible adaptative device that could be used on different CAN buses. Therefore we need to be able to modify some of its configuration, mainly baud rate.

To achieve that, we will have to create a communication protocol between the PC and the DAQ module. The esiest way is to use a command based communication. The packet frame would contain the command code and data related to the command.

Captures should be saved to SD card on files (file system).

Here is a possible list of commands that could be implemented :

* Set the device Baudrate
* Read the device Baudrate
* Enable/Disable capture (broadcasting Rx CAN packets to the network)
* Set Rx Port (optional)
* Set IP address (optional)
* Rx Message (message received)
* Tx Message (transmit a message)
* Set filters (optional)
* Ping (optional)
* Fetch traces (from SD card)
* Erase traces (in SD card)
* Get error count
* Get bus state (heavy, …)
* Reset captures (timesamp, numbering)
* Device resetted (device was reset)

When capture is enabled, each CAN message will be broadcasted on the network with the following information :

**message number (4 bytes, uint32) | timestamp (4 bytes, uint32, tenth of ms) | CAN msg ID (4 bytes, uint32) | DLC (1 byte, uint8) | Data (8 bytes, padding of 0 if DLC < 8)**

The commands payloads and details will be defined in a specific header file.

**About the firmware tasks**

Here are some priority of the firmware :

* Set the message timestamp (as fast as possible, interrupt ?)
* Send the message over Wifi
* Write the message to trace file (SD card)

Notes :

* Writing to SD card might be slow, so it must not be done in an interrupt to avoid blocking the program.
* The most priority is to set the message timestamp so it is as precise as possible.
* The trace files as binary files will take much less time to process and be smaller size than text files