**FSAE Datalogger Overview**

Car operates as a system composed of 4 interlinked subsystems.

1 The physical system which are the interlinked mechanical components that create and govern the motion of the card the inputs are physical from the driver and its outputs are analog + digital + frequency signal output to the datalogger.

2 The driver which pulls environmental input and converts to physical output to the physical system (ie. steering wheel, throttle.

3 The engine/transmission/turbo control electronic system (Haltech) that governs the proper running of the engine and feeds sensor output data. The input data to the engine control is from the physical system (throttle), environmental (ie. MAF), or feedback input from itself.

4 The Arduino microcontroller cluster. This includes the datalogger and the dash panel. This is the part that provides some limited feedback to the driver through the dash and collects information from the physical and engine control systems for tuning the system between uses.

The subsystem laid out in this documentation will be the datalogger.

**The Datalogger**

The datalogger 2018rev. is implemented using a shield, specifically the Seeedstudio CAN-BUS Shield V2.0 <http://wiki.seeedstudio.com/CAN-BUS_Shield_V2.0/>. This shield plugs into the Arduino allowing passthrough to many of the Arduino’s inputs and outputs. This shield has a SD card slot that allows for saving data of interest.

Inputs to the datalogger can be divided into analog signals, digital signals, and frequency signals. Each type of signal has its own bank of inputs through which it transmits real time sensor data to be captured sequentially by the Arduino and written to the SD card.

Analog signals: This includes suspension potentiometers??, \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_. These signals are first fed into a multiplexer and then each separate signal is read by the Arduino using selects???

Digital signals: This includes gps???, orientation sensor???, \_\_\_\_\_\_\_\_\_\_. Each signal has its own input into a Can-Bus Shield/Arduino port.

Frequency signals: This includes the wheel-speed sensor(s??). Input into shield???

**Important sub-guides (ie. Connecting the datalogger to the haltech)**

**Haltech to Datalogger:**

Connecting via the serial port. Baud rate of the haltech is \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ therefore sampling??? baud rate of the Arduino should be???.

**Analog sensor to Multiplexer:**

Jasalfjaljsfljasljkl\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Multiplexer to Datalogger:**

Jafsl;fjasjflajfljal\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Digital sensor to Datalogger:**

Js;fjalsjflasjflaj\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_--

**Frequency sensor to Datalogger:**

Ajsldfjalsjfdlajslfja\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Important component specs???? (Voltages?? Baud rates??? Filter Bandwidths??? Noise Frequencies??? Etc???)**

As;fjlajslfjasfjlajf

Asjlfdj;alsjfldjalfj;;a

Ajsl;fjl;ajsofpjwojeoruoa

Apwueoruqpowurq

Nzvnxznvppwjfpajs

**Sample Code??? Or code Functions overview (supercomments??)?? Or definitions of library functions used???**

Ajs;fljasjdfla;

Ewprupowquwrwqpowru

Fnvpnpznxkvnzlnkxvnz

N;lfjlsjg;ljfljasf;alfjasd

GPS debugging baud rate 115200 can switch in setup loop to 9600 (more common). GPS uses pins 2 and 3 currently not Tx Rx. Tx -> 2, Rx -> 3.

GPS eventually on Tx Rx, but very useful for debugging. (They go straight to USB).