**CODE REFACTOR OVERVIEW:**

Non-Boolean variables in throttle.c will be split into two structs:

-RawSensorValues

-FilteredSensorValues

Accessing throttle data will be through structs. However, getter functions for pedal readings and faults remain the same.

**CHANGES:**

(1) Variables placed in created struct: **Raw\_sensor\_value**

* adcData\_t FP\_data\_array[3]
* unsigned int volatile BSE\_sensor\_sum;
* unsigned int volatile FP\_sensor\_1\_sum;
* unsigned int volatile FP\_sensor\_2\_sum;

(2) Variables placed in created struct: **Filtered\_sensor\_value**

* float BSE\_previous\_filtered\_sensor\_values;
* float APPS1\_previous\_filtered\_sensor\_values;
* float APPS2\_previous\_filtered\_sensor\_values;
* float volatile Percent\_APPS1\_Pressed;
* float volatile Percent\_APPS2\_Pressed;
* float FP\_sensor\_diff;
* float volatile Percent\_BSE\_Pressed;

(3) Merged:FP\_data\_ptr & FP\_data-> **FP\_data\_array**

* This way we won’t need will to initialize FP\_data\_ptr value to FP\_data once a RawSensorValues variable is defined (or else compiler initializes FP\_data\_ptr to null, which would create problems when passing structs through the queue pipeline).
* C arrays are already pointers, so we remove any ambiguity.

(4) Moved 5 extern declarations into throttle.h

* extern TimerHandle\_t xTimers[NUMBER\_OF\_TIMERS];
* extern bool APPS1\_RANGE\_FAULT\_TIMER\_EXPIRED;
* extern bool APPS2\_RANGE\_FAULT\_TIMER\_EXPIRED;
* extern bool BSE\_RANGE\_FAULT\_TIMER\_EXPIRED;
* extern bool FP\_DIFF\_FAULT\_TIMER\_EXPIRED;

(5) Removed 7 unused variable declarations/definitions

* float BSE\_filtered\_sensor\_value;
* float APPS1\_filtered\_sensor\_value;
* float APPS2\_filtered\_sensor\_value;
* uint32\_t volatile fault\_10DIFF\_counter\_ms = 0;
* uint32\_t fault\_BSE\_Range\_counter\_ms = 0;
* uint32\_t fault\_APPS1\_Range\_counter\_ms = 0;
* uint32\_t fault\_APPS2\_Range\_counter\_ms = 0;

(6) Moved 5 variable definitions from the global scope into their respective local function scope

* bool is\_there\_10DIFF\_fault;
* bool is\_there\_BSE\_range\_fault;
* bool is\_there\_APPS1\_range\_fault;
* bool is\_there\_APPS2\_range\_fault;
* bool is\_there\_brake\_plausibility\_fault;