USING THE MATLAB TOOLSET TO IMPROVE EFFICIENCY IN THE EOBD CALIBRATION PROCESS

N Wright and R Drane, Ford Motor Company Limited, UK

Abstract

There is a need in the Motor Industry to reduce Product Development cycle time and drive down development costs by reducing the amount of testing required and reducing the reliance on expensive prototype vehicles. In order to achieve this goal new innovative processes must be developed with increased use of CAE and desktop calibration. Ford Motor Company with the Mathworks has developed a suite of tools to enable reduced testing, increased proportion of desktop calibration and a significant increase in test data reusability within the EOBD calibration process.

The new process makes use of Simulink strategy models with a user friendly front end GUI to enable easy desktop calibration and validation, a new data editor tool to enable fast batch processing of recorded calibration data, the use of the MBC toolset to develop accurate surface models of key EOBD control system characteristics and CAGE to populate calibration parameters from the MBC models.

This paper will give an overview of the process and tools giving some practical examples of their application.

Introduction

With the exception of a few selected products, manufacturers have to develop new products on a regular basis as the demands of the marketplace regularly change and so products need to be continuously improved and updated to enable the necessary volumes of sales to justify their profitable manufacture to be achieved. Figure 1. shows a typical post launch product life cycle.

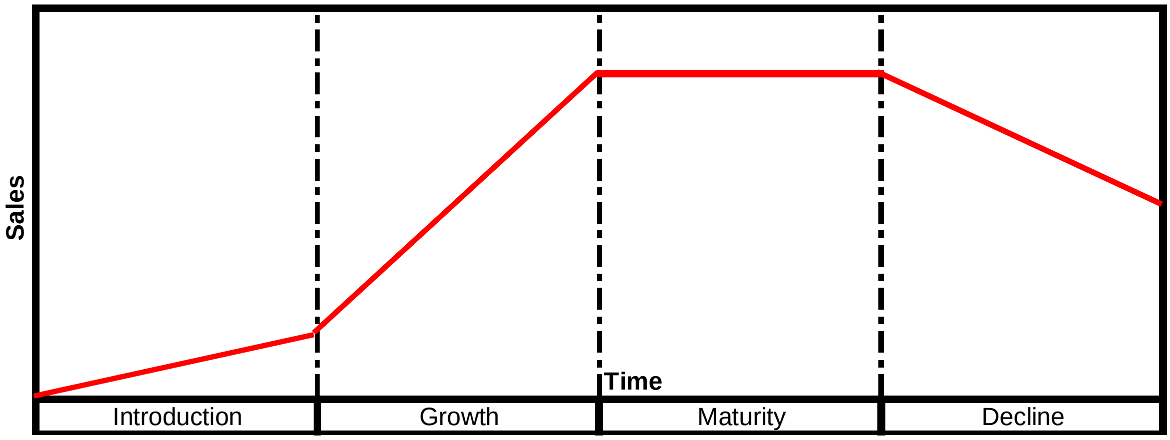


Figure 1.

The challenge faced by most manufacturing companies lies not only with accelerating the period of initial sales growth after launch and maintaining the period of sales maturity for as long as possible, but also reducing the costly development time before launch.

For companies manufacturing high technology products dependant on significant R&D effort and where the market is heavily influenced by rapid technological change, the ideal life cycle is difficult to achieve. In the worst case the development time is long and the development costs are high; the induction growth period is long and shallow; the maturity period is short and the decline is fast. In this situation the return on investment is marginal with some income barely exceeding expenditure. This situation is illustrated in Figure 2.

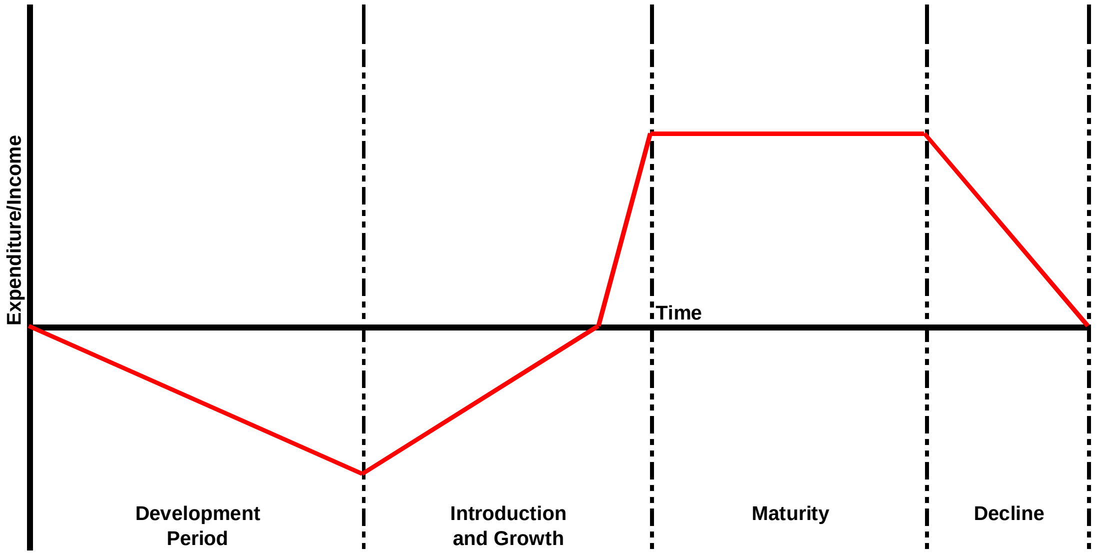


Figure 2.

A major aim of most companies is to develop new products using a process that mimics the ideal life cycle as closely as possible. Having a world class product which clearly addresses the identified needs of the market place is one means of theoretically ensuring that, after a product launch, growth of sales is rapid and the highly profitable maturity period is extended for as long as possible. The increasingly competitive global market has focused attention on getting new products to launch quickly and cheaply, with the main tool for achieving this being the reduction of development time and costs.

The EOBD calibration and validation process has historically been heavily reliant on using expensive prototype vehicles and facilities for development and sign-off, much of the testing is repetitive and time consuming, additionally the test data produced has limited usefulness. To achieve the goals outlined above the development process needs to be more efficient, testing needs to be reduced without compromising calibration robustness and test data needs to be reusable throughout the development process.

Identifying the Opportunity

Within Ford Motor Company an increasing emphasis has been placed on reducing the number of prototype vehicles and increasing the usage of CAE tools and methodologies. Identifying suitable areas of the EOBD calibration development process where efficiencies could be made through the use of CAE methodologies was the first task and a number of areas were quickly identified.

- Tasks that require repetitive testing, often under controlled climatic conditions to achieve an optimum calibration.
- Tasks that require a large amount of manual data post processing.
- Tasks that require system characterisation over a wide range of conditions.
- Tasks where a system model could be developed to populate the calibration.

- Tasks where recorded data could potentially be re-used to validate consequent calibration changes.

Ford has a long standing working relationship with the Mathworks and so the MATLAB suite of tools was used as a starting point and a number of custom built tools were developed and enhanced as experience and confidence in the new methodologies grew. The following tools have been developed and integrated into the EOBD calibration development and validation process.

Simulink based strategy models - Areas of the PCM strategy have been modelled using Simulink to enable desktop calibration and validation of Catalyst, HEGO, Misfire and Fuel Monitors. Models have a user friendly front end GUI that enables novice MATLAB users to be able to work with the tools.

Custom Data Processor – Tool developed that enables rapid automated data manipulation and processing.

MBC Toolbox – Tools used to characterise systems and develop accurate surface models of EOBD monitor inputs. Models can be used to develop EOBD monitor thresholds and predict changes to thresholds based on changes to calibration of monitor inputs.

CAGE – Tool used to populate calibration tables based on models developed using MBC.

Using the Toolset – EOBD HEGO Monitor

Monitoring the signal from the HEGO sensor consists of a number of diagnostics that check the performance of the sensor under a wide range of climatic and engine conditions. Parts of the calibration are performed under stable ambient and engine conditions that can only be maintained for a short period of time. Historically the calibration consisted of repetitive testing with an engine stabilisation period between tests. To complete the calibration and validation process could take a number of days utilising an expensive climatic chamber.

To improve this process a tool was developed that required data from one test with the calibration being automatically generated by a tool that utilised a Simulink model of the strategy and a batch processing tool that would determine the optimum settings for the calibration. A comparison of the two processes can be seen in Figure 3.

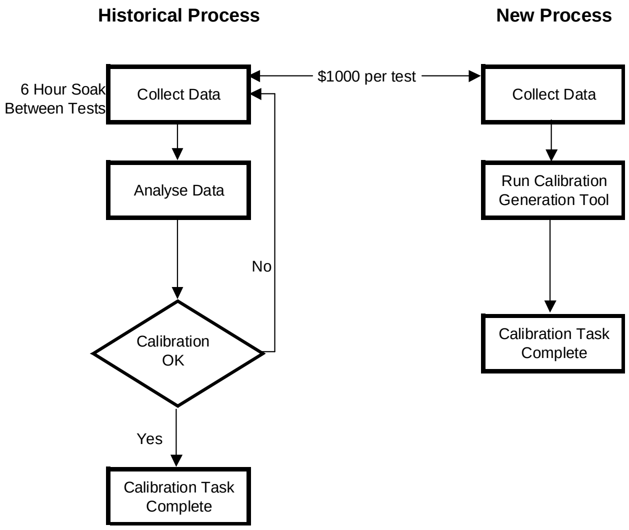
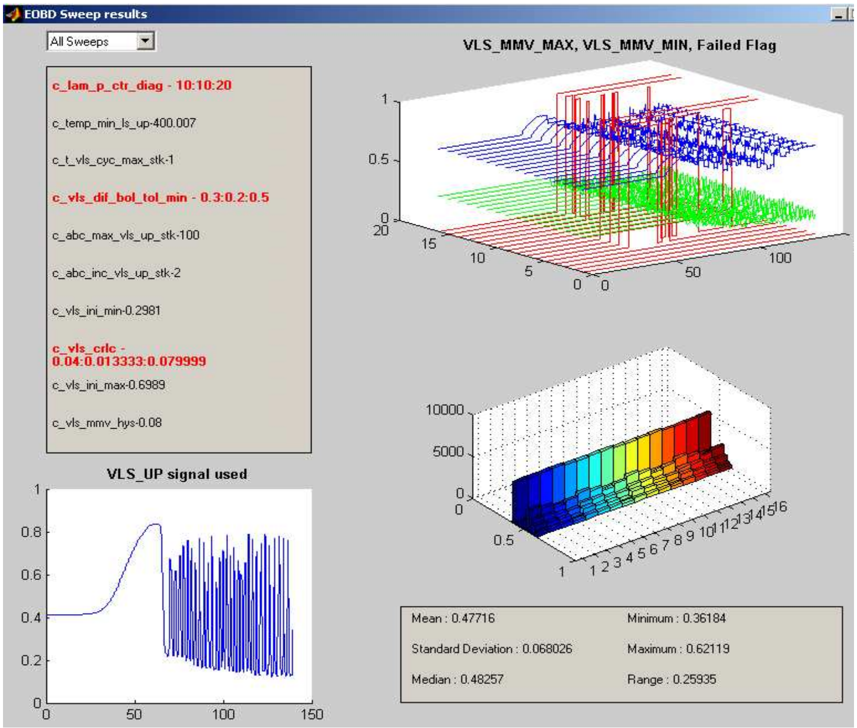


Figure 3.

This area of the calibration is highly dependent on the sensor specification, the sensor package and the closed loop air fuel control calibration. As a result there is usually a number of iterations required using the historical process before the calibration is optimised. The new process uses the one set of original data and the calibration generation tool automatically determines the optimum calibration. An example of the output of the tool can be seen in Figure 4.



c_lam_p_ctr_diag - 10:10:20

c_temp_min_ls_up-400.007

c_t_vls_cyc_max_stk-1

c_vls_dif_bol_tol_min - 0.3:0.2:0.5

c_abc_max_vls_up_stk-100

c_abc_inc_vls_up_stk-2

c_vls_ini_min-0.2981

c_vls_crlc - 0.04:0.013333:0.079999

c_vls_ini_max-0.6989

c_vls_mmv_hys-0.08

Mean : 0.47716
Minimum : 0.36184
Standard Deviation : 0.068026
Maximum : 0.62119
Median : 0.48257
Range : 0.25935

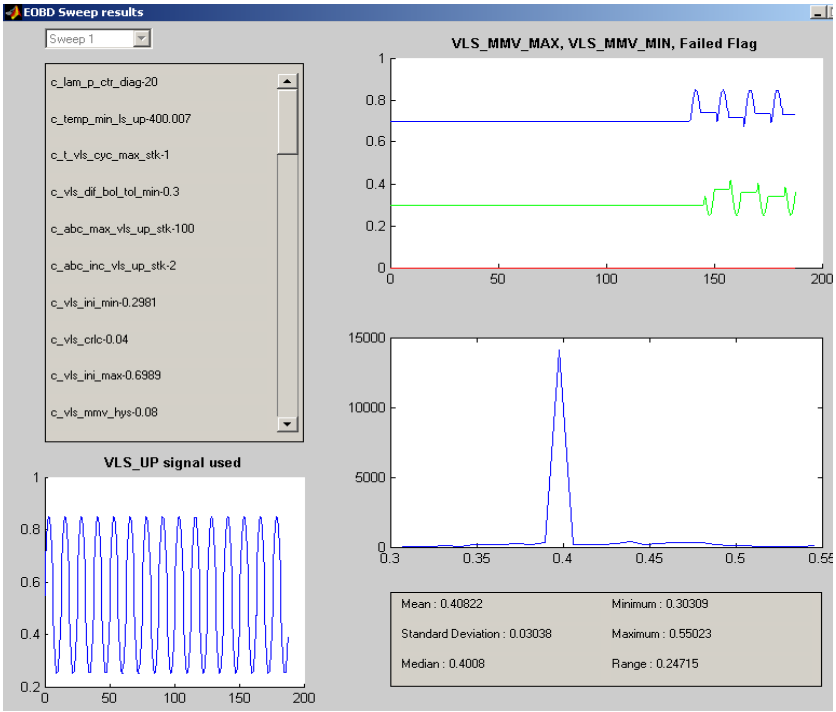
Figure 4.

In the historical process after the optimum calibration is derived, calibration robustness is tested by using limit sensors on the vehicle and repeating the testing in the climatic chamber. In the new process calibration robustness can be tested on the desktop using the tool's built in signal generator. An example can be seen in Figure 5.

When the threshold for the monitor is dependent on a surface rather than a single point, additional tools are required to improve the process efficiency. The response of the input to engine conditions is traditionally mapped on a chassis dyno across the monitor operating range. The input response is also dependent on a number of calibration variables. If the critical calibration parameters change then the input response must be mapped again. To support the calibration development process the input response may have to be mapped a number of times over the length of the program.

The new process uses the MBC toolset to produce a surface model of the input response that is derived from the engine conditions and the key calibration variables. Using this method if the calibration variables change the model can be used to re-calculate the threshold surface without the need to re-map the response.

As the surface is developed from an MBC model rather than from a manual mapping process, time can be saved during the mapping process. A new data processing tool is used to automate the data post processing, the CAGE tool is used to auto generate the calibration tables and normalising functions while a Simulink Simulation tool is used to validate the calibration. A comparison of the two processes and the tools used can be seen in Figure 6.



c_lam_p_ctr_diag-20
c_temp_min_ls_up-400.007
c_t_vls_cyc_max_stk-1
c_vls_dif_bol_tol_min-0.3
c_abc_max_vls_up_stk-100
c_abc_inc_vls_up_stk-2
c_vls_ini_min-0.2981
c_vls_crlc-0.04
c_vls_ini_max-0.6989
c_vls_mmv_hys-0.08

Mean : 0.40822
Minimum : 0.30309
Standard Deviation : 0.03038
Maximum : 0.55023
Median : 0.4008
Range : 0.24715

Figure 5.

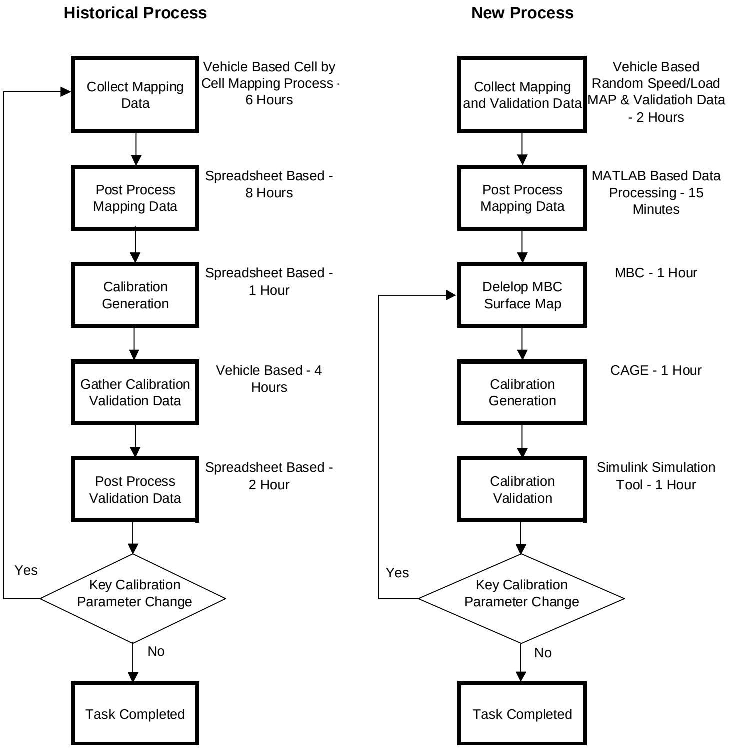


Figure 6.

It can be easily seen that the new process reduces the once through process and vehicle usage time and any subsequent runs are limited to a desktop process. The key to this process is the development of the MBC model. It is extremely important to ensure that the model inputs are chosen such that the robustness of the model allows a high degree of confidence in thresholds developed without vehicle validation. The ease of use and the statistical analysis within MBC gives the user the opportunity to accurately evaluate, many iterations of the model inputs quickly. The validation of the model inputs is proven when it can be applied across a wide range of powertrain applications without modification. Figure 7. shows the results from the model developed for this application when applied across a number of powertrain combinations.

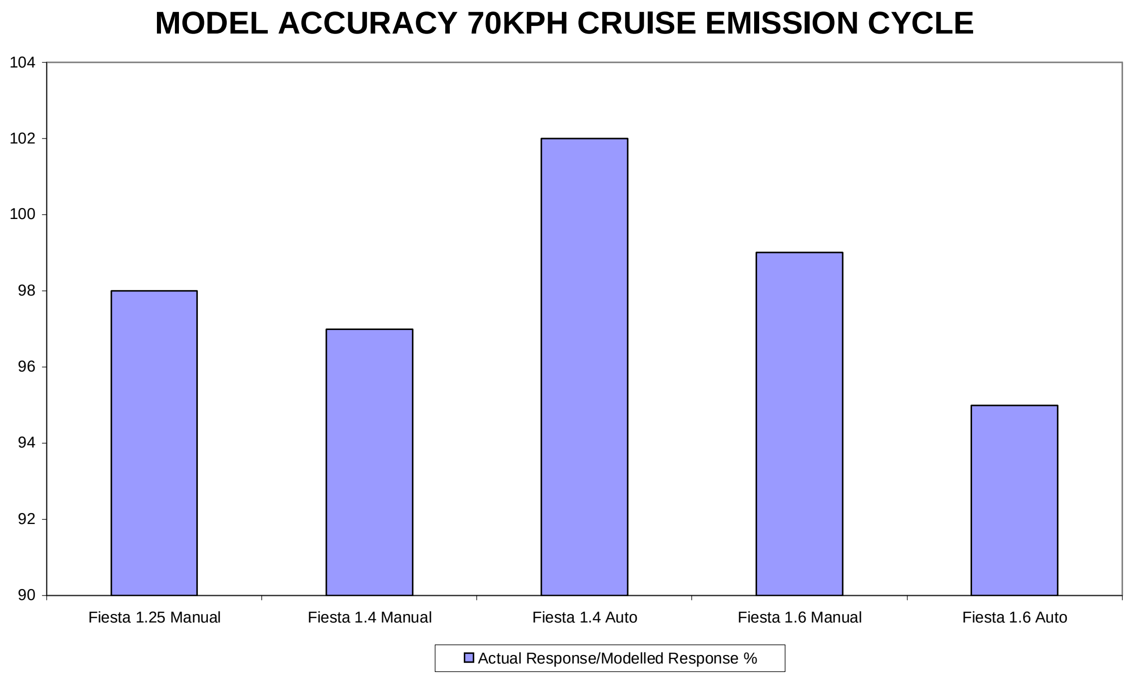
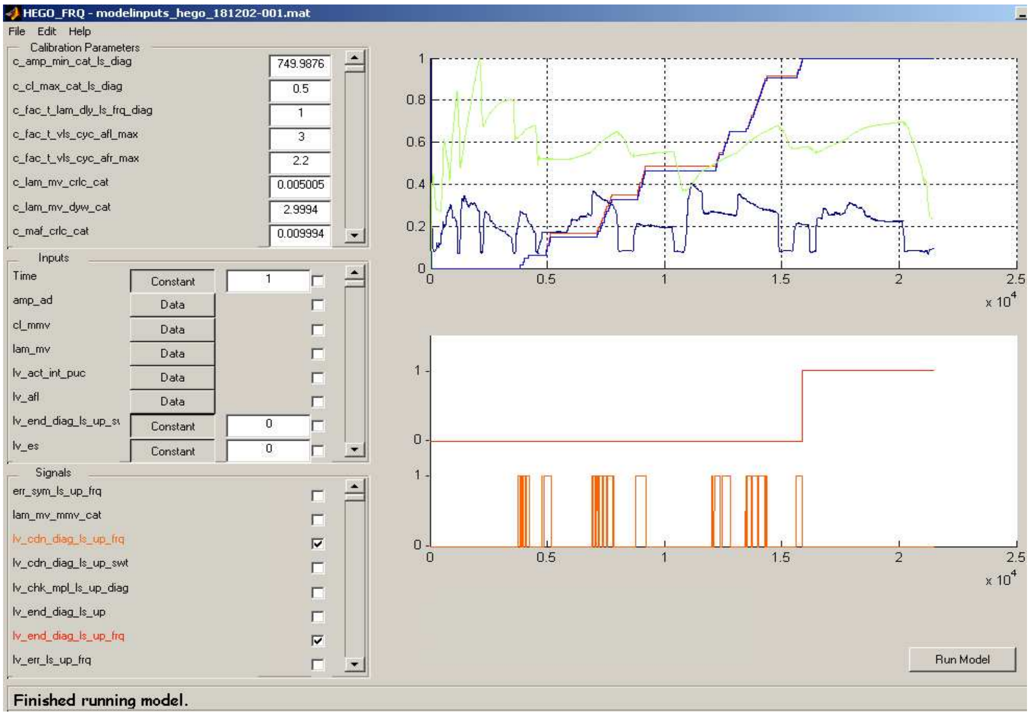


Figure 7.

The tools used in the process described above have many other applications in the development process. The Simulink Simulation tools in particular allow the data collected to be re-used to validate subsequent calibration changes. Data is collected on chassis rolls and on the road throughout the development process. Historically if calibration changes are made then these tests need to be repeated in order to validate the changes. With the new process if a calibration change is required the original road and chassis roll data can be run through the Simulink model in a desktop environment to determine the effects of an proposed changes. In the example shown in Figure 8. it was found that the operating window of the monitor needed to be extended due to interactions with other areas of the calibration affecting the monitor completion. The original data collected on the road was run through the Simulink model with and without the proposed changes. It could be seen that the changes allowed the monitor to complete easier with no adverse effect on the accuracy of the model prediction. Therefore the calibration changes were validated desktop without the need to re-validate the calibration on the road.

BEFORE CALIBRATION CHANGE



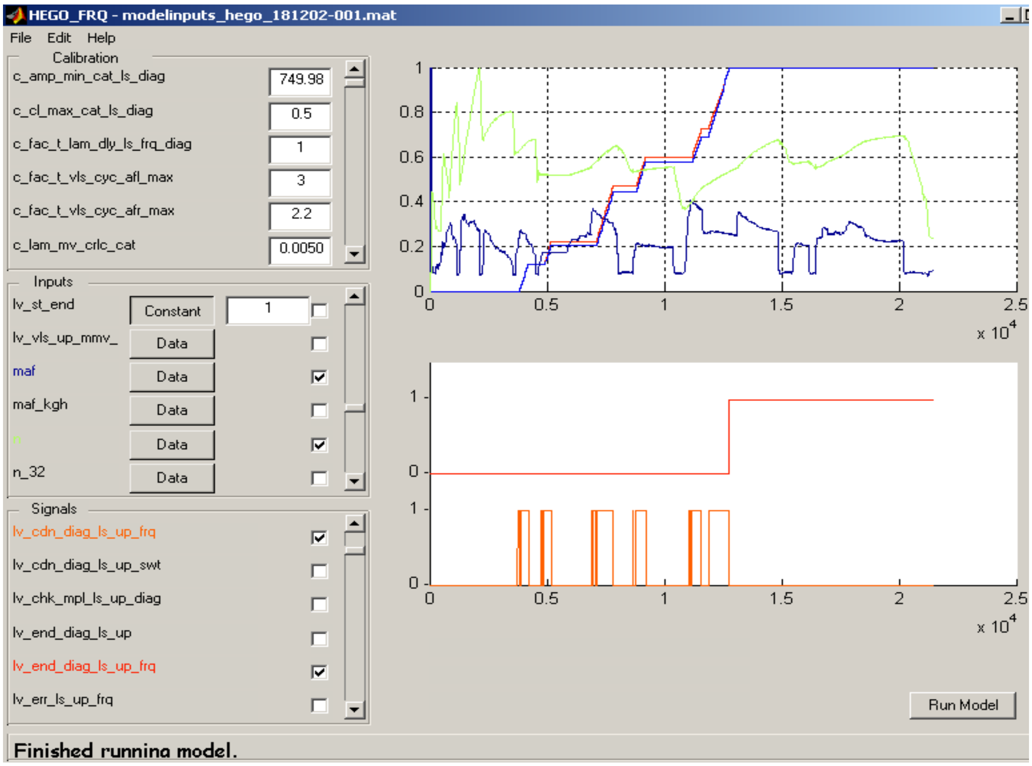
Calibration Parameters
c_amp_min_cat_ls_diag 749.9876
c_cl_max_cat_ls_diag 0.5
c_fac_t_lam_dly_ls_frq_diag 1
c_fac_t_vls_cyc_afl_max 3
c_fac_t_vls_cyc_afr_max 2.2
c_lam_mv_crlc_cat 0.005005
c_lam_mv_dyw_cat 2.9994
c_maf_crlc_cat 0.009994

Inputs
Time Constant 1 ☐
amp_ad Data ☐
cl_mmv Data ☐
lam_mv Data ☐
lv_act_int_puc Data ☐
lv_afl Data ☐
lv_end_diag_ls_up_sv Constant 0 ☐
lv_es Constant 0 ☐

Signals
err_sym_ls_up_frq ☐
lam_mv_mmv_cat ☐
lv_cdn_diag_ls_up_frq ☑
lv_cdn_diag_ls_up_swt ☐
lv_chk_mpl_ls_up_diag ☐
lv_end_diag_ls_up ☐
lv_end_diag_ls_up_frq ☑
lv_err_ls_up_frq ☐

Finished running model.

AFTER CALIBRATION CHANGE



Calibration
c_amp_min_cat_ls_diag 749.98
c_cl_max_cat_ls_diag 0.5
c_fac_t_lam_dly_ls_frq_diag 1
c_fac_t_vls_cyc_afl_max 3
c_fac_t_vls_cyc_afr_max 2.2
c_lam_mv_crlc_cat 0.0050

Inputs
lv_st_end Constant 1 ☐
lv_vls_up_mmv_ Data ☐
maf Data ☑
maf_kgh Data ☐
n Data ☑
n_32 Data ☐

Signals
lv_cdn_diag_ls_up_frq ☑
lv_cdn_diag_ls_up_swt ☐
lv_chk_mpl_ls_up_diag ☐
lv_end_diag_ls_up ☐
lv_end_diag_ls_up_frq ☑
lv_err_ls_up_frq ☐

Finished running model.

Figure 8.

Using the Toolset – EOBD Misfire Monitor

The purpose of the EOBD misfire monitor is to detect the occurrence of misfire at a rate that is sufficient enough to cause an emission related threshold violation or to cause heat damage to the catalytic converter that resides in the exhaust system. Misfire is deemed to be a total loss of torque output from a cylinder during the combustion cycle.

A typical inexpensive method of detecting misfire that is widely used within the auto industry is to infer the torque loss due to misfire through an evaluation of crankshaft motion, usually acceleration variation. However, crankshaft acceleration variations occur frequently under normal engine operating conditions; examples include transient conditions (e.g. throttle movements), deceleration, deliberate torque reduction actions to improve driveability and the transmission of driveline shock to the crankshaft from the road (rough road). Thus, for a misfire detection system to be accurate and effective, it must be capable of detecting true misfire, but at the same time must be capable of ignoring all the instances where the crankshaft acceleration behaves like a misfire has occurred, but is in fact normal engine operation.

Misfire algorithms typically reconstitute the crankshaft acceleration in some way in order to emphasise the differential of the crankshaft motion signal between misfire and normal combustion. A typical crankshaft motion signal with a low misfire rate is shown in Figure 9. The peaks under misfire infer torque loss and these are mapped across the entire engine operating range. The algorithms also apply disablement features for conditions where misfire detection is deemed unreliable or the torque is fluctuating due to normal engine operating conditions. Disablement features are typically active at low engine load and torque, during forced torque transitions, during transient conditions and during rough road conditions. Misfire detection thresholds are determined from the torque loss map, as are the calibrations for some of the algorithm's internal detection filters and some of the disablement features. Thus, due to the various dependability's, determination of the final misfire detection calibration is not a trivial task.

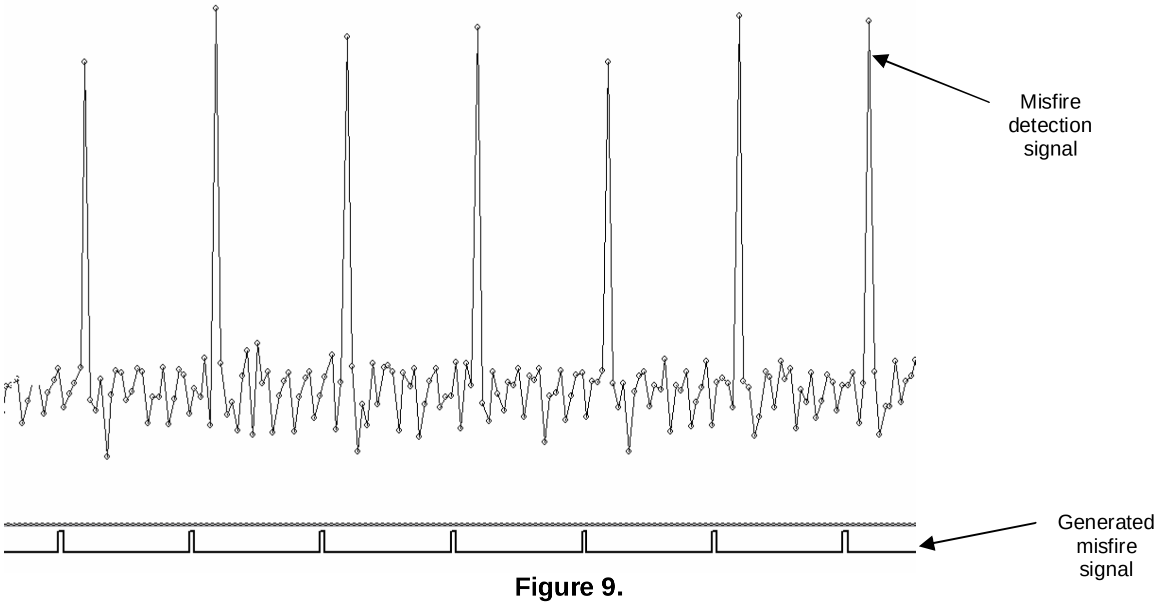


Figure 9.

Historically, the determining of the misfire detection calibration has been a vehicle and test facility intensive process that requires a number of iterations to obtain optimal performance. Figure 10 compares the historical process with a new process that utilises the Matlab toolset.

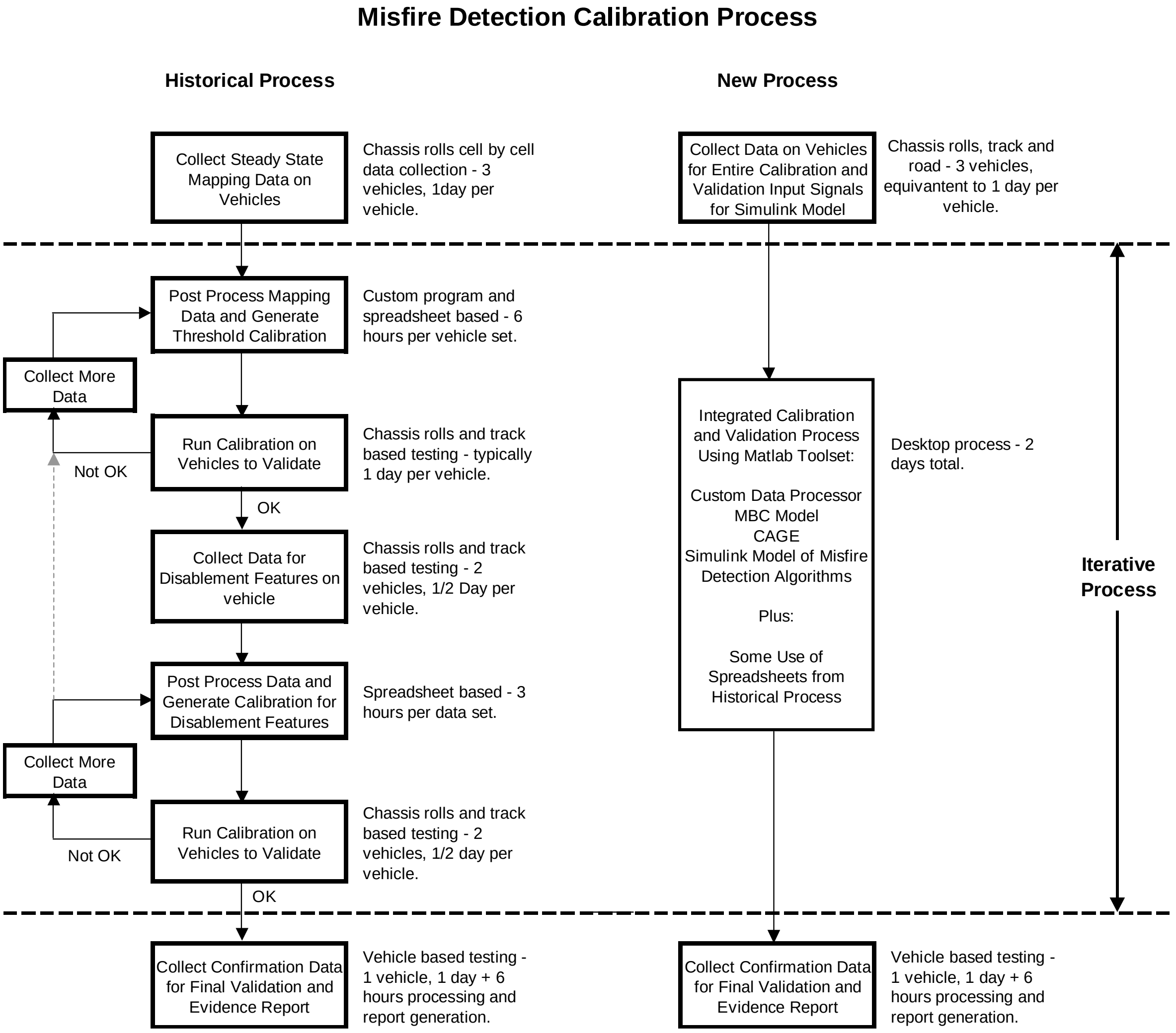


Figure 10.

Both processes require actual vehicle data to be collected, but the Matlab toolset offered the opportunity to reduce costs and calibration development time by moving the iterative process of data analysis, calibration and validation away from the vehicle to the desktop. Not only is the Matlab toolset process quicker, it also makes substantial untold savings by eliminating the time required to find vehicles for test, scheduling facility time and paying facility costs. This is particularly valuable in an environment where vehicles and facilities are already highly utilized for other tasks. The Matlab toolset process is very flexible and in each case so far has proved to be a robust process that has generated misfire detection calibrations with a high degree of confidence prior to the collection of the confirmation data.

Detail of the Matlab Toolset Integrated Calibration and Validation Process is shown in Figure 11.

Data Collection

The data collection process is slightly different than for the historical process. In the historical process, data is collected at various times as the calibration process progresses. Steady state data is collected from vehicles on chassis rolls where the engines are subjected to various rates of forced misfire. Data is

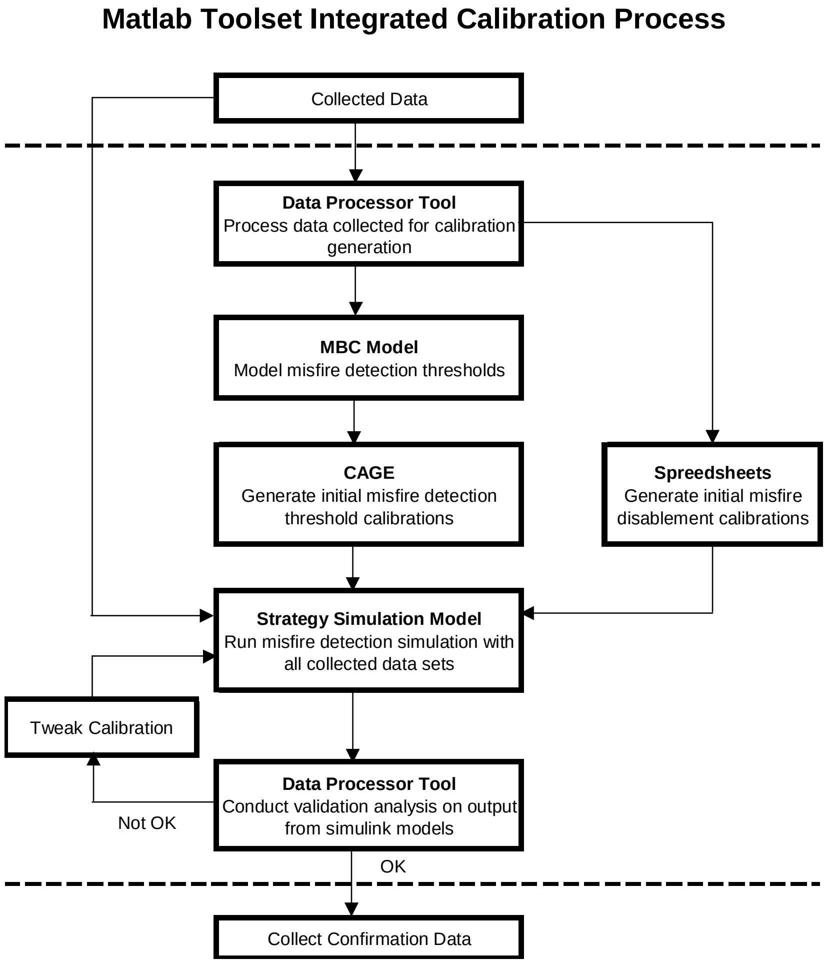


Figure 11.

used to map the misfire characteristic across the full misfire operating range on the engine, with each data collected point targeted to provide the calibration for a specific box in the calibration table. Thus precise setting of engine speed and load conditions for each data point is essential to establish accurate values for the tables. Obtaining precise speed/load conditions, even with a throttle control device, is costly on time. Later on, more data of a similar nature is collected for validation and refinement purposes. Once the steady state misfire characteristic is established the disablement features are calibrated using data already collected from the steady state testing and from further vehicle data collected on road and track. Once the misfire disablement calibrations are established, further vehicle based testing is required to validate them.

In the new process, all the data required to generate the complete calibration is recorded at the start of the process. Similar types of vehicle based data sets are recorded on chassis rolls, track and road as the historical process, except that many more parameters are recorded. These parameters are required for generation of the calibration as before, but additional parameters are also required to run the simulink model later on. The additional parameters provide engine operating conditions and misfire detection input information that is unaffected by the misfire calibration, such as engine speed, load, torque, ignition timing, crankshaft motion timing, etc. Additionally, since MBC Model and CAGE are used to generate the steady state misfire characterisation tables, precise setting of engine speed/load conditions is not required, thus making the steady state data collection phase substantially quicker.

Data Processing

The custom developed processing tool is used to analyse and manipulate the recorded data. This is a very flexible and powerful tool that can be configured to perform a variety of statistical actions and analytical functions. It is built into the Matlab toolbox environment and so has all the toolbox functions available for it to use. Some unique features of this tool include:

a) Displaying of signals to be analysed.
b) Enables use of user defined customized functions.
c) Data manipulation enables:
   i) Filtering out unwanted data from a data file and analysing the rest.
   ii) Analysing data based on a condition of a flag either coincidently or displaced by a specified number of events.
d) A large number of individual files can be batch processed in a matter of minutes
e) Results of the data processing can be stored in Matlab format for direct import into MBC Model and in .csv format for import into spreadsheets.
f) Different analytical tasks for different projects can be saved as templates.
g) It can be linked to simulink models to analyse the outputs of the simulations.

For the misfire analysis, the tool is used to batch process 100 plus files at a time. Not only does it provide basic statistical information of some of the signals but it also performs more complex tasks where the misfire signal needs to be separated into portions based on misfire or normal combustion. As an example, referring to Figure 9, the signal to noise ratio of the misfire detection signal is of particular interest where:

$$\text{SNR} = \frac{\text{mean of misfires} - \text{mean of combustion}}{\text{std dev of misfires} + \text{std dev of combustion}}$$

Note that the misfire detection signal is displaced from the generated misfire signal by a few combustion events. The data processing tool can handle this situation quite effortlessly.

Calibration Generation and Validation

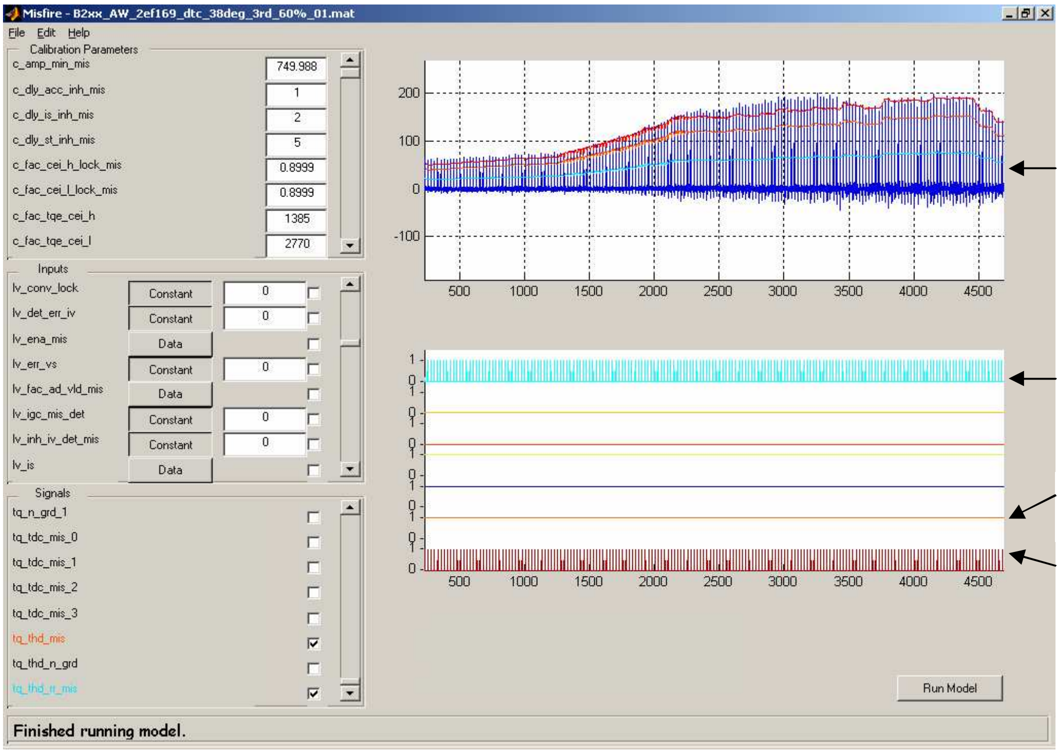
The analysis results generated by the data processing tool are imported into MBC Model and then to CAGE to generate maps of the misfire detection thresholds and into spreadsheets to help generate the disablement feature calibrations.

Using a front end GUI, the calibrations are entered into a simulink model of the misfire detection algorithms. The model is run using input signals collected at the same time as the original calibration data, thus eliminating the need to test more vehicles. The simulation for each file used can be viewed individually on the GUI, or be linked directly with the data processing tool for batch processing. A different template in the data processing tool is used to the one for calibration generation. Between the simulink model and the data processor, the calibration can be easily massaged at the desktop until the desired validation data is generated.

Examples of the simulink front end GUI and simulation results are shown in Figure 12.

Once the desktop process is complete, confirmation data is collected as per the historical process to produce evidence reports.

60% Crowd through engine speed range:



Misfire - B2xx_AW_2ef169_dtc_38deg_3rd_60%_01.mat

Calibration Parameters

c_amp_min_mis	749.988
c_dly_acc_inh_mis	1
c_dly_is_inh_mis	2
c_dly_st_inh_mis	5
c_fac_cei_h_lock_mis	0.8999
c_fac_cei_l_lock_mis	0.8999
c_fac_tqe_cei_h	1385
c_fac_tqe_cei_l	2770

Inputs

lv_conv_lock	Constant	0	☐
lv_det_err_iv	Constant	0	☐
lv_ena_mis	Data		☐
lv_err_vs	Constant	0	☐
lv_fac_ad_vld_mis	Data		☐
lv_igc_mis_det	Constant	0	☐
lv_inh_iv_det_mis	Constant	0	☐
lv_is	Data		☐

Signals

tq_n_grd_1	☐
tq_tdc_mis_0	☐
tq_tdc_mis_1	☐
tq_tdc_mis_2	☐
tq_tdc_mis_3	☐
tq_thd_mis	☑
tq_thd_n_grd	☐
tq_thd_n_mis	☑

Finished running model.

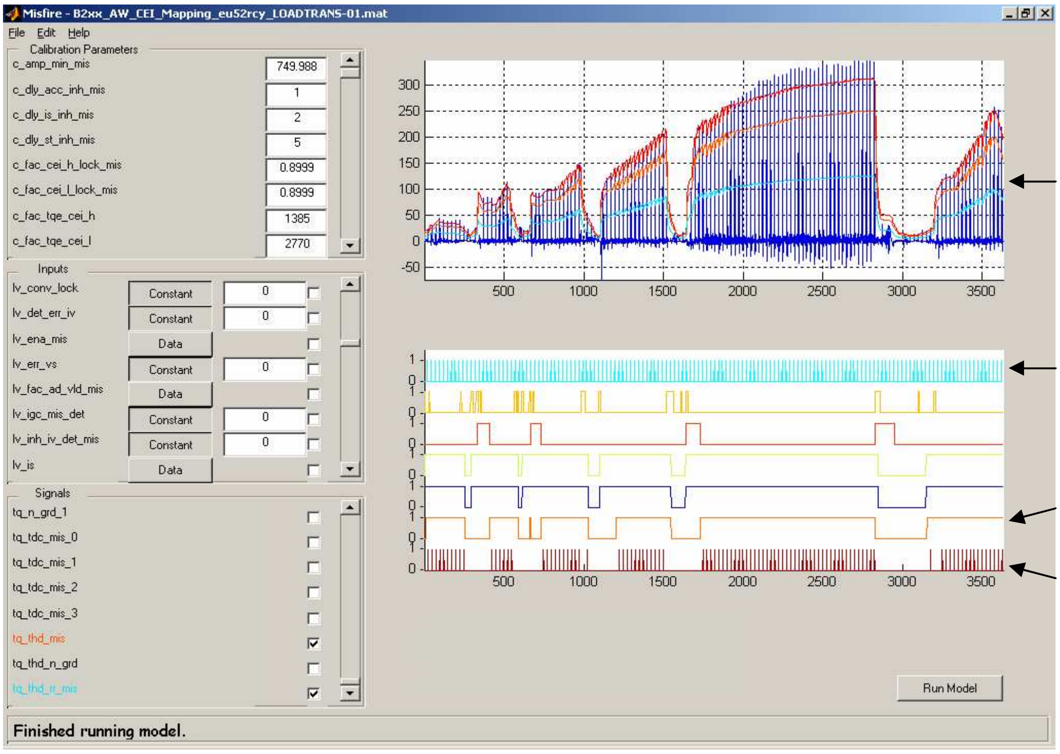
Misfire signal and thresholds

Misfire generated

Misfire enabled flag

Misfire detected

Transients:



Misfire - B2xx_AW_CEI_Mapping_eu52rcy_LOADTRANS-01.mat

Calibration Parameters

c_amp_min_mis	749.988
c_dly_acc_inh_mis	1
c_dly_is_inh_mis	2
c_dly_st_inh_mis	5
c_fac_cei_h_lock_mis	0.8999
c_fac_cei_l_lock_mis	0.8999
c_fac_tqe_cei_h	1385
c_fac_tqe_cei_l	2770

Inputs

lv_conv_lock	Constant	0	☐
lv_det_err_iv	Constant	0	☐
lv_ena_mis	Data		☐
lv_err_vs	Constant	0	☐
lv_fac_ad_vld_mis	Data		☐
lv_igc_mis_det	Constant	0	☐
lv_inh_iv_det_mis	Constant	0	☐
lv_is	Data		☐

Signals

tq_n_grd_1	☐
tq_tdc_mis_0	☐
tq_tdc_mis_1	☐
tq_tdc_mis_2	☐
tq_tdc_mis_3	☐
tq_thd_mis	☑
tq_thd_n_grd	☐
tq_thd_n_mis	☑

Finished running model.

Misfire signal and thresholds

Misfire generated

Misfire enabled flag

Misfire detected

Figure 12.

Conclusions

The process and toolset described above is not confined to EOBD calibration applications. The process is very generic and can be adapted to suit a variety of calibration areas. The controller for the Simulink modes is very flexible and can be linked to any Simulink model that is configured correctly. The data post processing tool can be used to evaluate data collected from any calibration process and being linked to the MBC toolset can perform a wide range of data manipulation and interrogation. Simulink simulation tools have also been used to develop and validate calibrations for exhaust gas temperature models as well as FMEM calibrations for coolant temperature, MAP and MAF sensors. Used as a package the toolset described has enabled a significant saving in time and vehicle usage as well as allowing test data to be re-used.

Ford is exploring other areas where such tools can be utilised including ETC Safety and areas of Diesel calibration. Further use of CAE tools is the only way that Ford can meet its efficiency targets and reduce its calibration development span.