CRANFIELD UNIVERSITY

KARTHIKK RAJA GOPALAKRISHNAN

IVHM ON UAV FUEL SYSTEM TEST RIG

SCHOOL OF ENGINEERING
Aerospace Vehicle Design

MSc
Academic Year: 2010 - 2011

Supervisor: Dr. Suresh Perinpanayagam & Dr. Octavian Niculita
August 2011

CRANFIELD UNIVERSITY

SCHOOL OF ENGINEERING
Aerospace Vehicle Design

MSc

Academic Year 2010 - 2011

KARTHIKK RAJA GOPALAKRISHNAN

IVHM ON UAV FUEL SYSTEM TEST RIG

Supervisor: Dr. Suresh Perinpanayagam & Dr. Octavian Niculita

August 2011

This thesis is submitted in partial fulfilment of the requirements for the degree of MSc

THESIS HEALTH WARNING

This thesis has been assessed as of satisfactory standard for the award of the Master of Science Degree in Aerospace Vehicle Design. This thesis covers the part of the assessment concerned with the Group Design project. Readers must be aware that the work contained is not necessarily 100% correct, and caution should be exercised if the thesis or the data it contains is being used for future work. If in doubt, please refer to the supervisor named in the thesis, or the Department of Aerospace Engineering.

ABSTRACT

Unmanned Air Vehicle (UAV) show great promise for a range of civilian and military applications, especially “dull, dirty or dangerous” missions such as air-sea rescue, coastal and border surveillance, fisheries protection and disaster relief. As the demand for autonomy increases, the importance of correctly identifying and responding to faults becomes more apparent, as fully autonomous systems must base their decisions solely upon the sensors readings they receive – as there is no human on board. A UAV must be capable of performing all the functions that would be expected from a human pilot, including reasoning about faults and making decisions about how to best mitigate their consequences, given the larger context of the overall mission. As these autonomous techniques are developed their benefits can also be realised in non-autonomous systems, as real-time aids to human operators or crew.

The IRP thesis purpose is to develop a novel approach to IVHM that combines diagnostic analysis such as Detection and Isolation Statistics and Advanced FMECA. The software tool used to provide a creative diagnostic design is called eXpress. This project complements the design phase of an Unmanned Aerial Vehicle (UAV) fuel system from an IVHM perspective.

Keywords:

- Fault Detection
- Fault Isolation
- Fault Diagnosis.

ACKNOWLEDGEMENTS

The author would like to take this opportunity to personally thank his supervisor, Dr Suresh Perinpanayagam, for his continuous encouragement and professional and personal support throughout the author's IRP.

The author would also like to take this opportunity to thank the staff from Department of Aerospace Engineering and the mentors, Dr Octavian Niculita from Integrated Vehicle Health Management (IVHM) Centre, for his innovative scientific advice and support throughout the IRP. Thanks also extended to the staffs and members, Department of Aerospace Engineering, Cranfield University for guiding me to take this innovative project.

Finally, the author would like to thank his families for their understanding, belief, patience and commitment through the whole year of academic studies in Cranfield University, especially his father Gopalakrishnan Guruswamy Naidu and his mother Prakashini Gopalakrishnan.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

APU	Auxiliary Power Unit
BIT	Built In Test
COTS	Commercial-Off the Shelf
CUM	Cumulative
CUM FP	Cumulative Fault Percentage
DIAGML	Diagnostic Modelling Language
DFD	Data Flow Diagram
EUL	Expected Useful life
FDI	Fault Detection and Isolation
FMECA	Failure Mode Effect and Criticality Analysis
HDM	Hybrid Diagnostic Modelling
IVHM	Integrated Vehicle Health Management
LRU	Line Replaceable Unit
MBR	Model Based Reasoner
MCTI	Mean Cost to Isolate
MCTR	Mean Cost to Repair
MTBF	Mean Time between Failure
MTTI	Mean Time to Isolate
MTTR	Mean Time to Repair

RUL	Remaining Useful Life
TFD	Total Fault Detection
TFD (I)	Total Fault Detection with Interference
UAV	Unmanned Aerial Vehicle

1 INTRODUCTION

1.1 Fault Detection and Isolation Background

Current Civil and Defence aviation sectors are being subjected to more complex problems due to the increased number of components and sub assembly utilisation in the system. Due to this, locating faults becomes increasingly hard. Fault detection and Isolation becomes a challenge for the present system. The maintenance schedule also increases due to the complexity and the main factors influencing the maintenance such as cost and time increase proportionally with the complexity of the system. In order to overcome these problems and to meet customer demands such as safety, reliability and maintainability of systems, various fault detection and isolation methodologies have been introduced, and have received a positive response within the industry for the practical application of these fault diagnostic methodologies. A well reduced maintenance cost and increased availability of the system can be achieved by a well defined reliable diagnostic technique. Appropriate diagnostic tool selection must be an important factor for an industry because it has the potential to reduce life cycle costs. On board fault diagnosis plays a crucial role in unmanned aerial systems due to the lack of human interface to perform the safety functions of the system.

1.2 IVHM Definition

In Common terms "*Integrated Vehicle Health Management is the transformation of system data into information to support operational decisions that result in minimised maintenance actions, improved readiness and availability, reduced redundancies, product life extension and improved environmental impact*"(Hobbs, 2009).

From the Maintenance aspect, "*IVHM is a comprehensive health management system philosophy which integrates the results from the monitoring sensors all the way through to the reasoning software that provides decision support for* *optimal use of maintenance resources*"(Vachtsevanos, 2006). It explains, if the failure effects are observed in a particular system, it is possible to predict the root cause of the failure.

With regards to safety, "*IVHM capabilities will enable the rapid detection and diagnosis of these adverse events (in both the hardware and the software) essential to the safe operation of the vehicle and will enable the estimation of the condition severity and the remaining useful life (RUL) with confidence bounds for the affected system(s)*"(Ashok Srivastava, 2008).

1.2.1 Objective of IVHM

While explaining the goal of IVHM, it must be stated that it brings maintenance cost, increased reliability and availability and increased product life time by utilising the transformed sensor values into information which is active(Hobbs, 2009).

In a report from aviation safety program of NASA, the goals of IVHM are defined as "*reduce system and component failures as causal and contributing factors in aircraft accidents and incidents. Provide continuous on-board situational awareness of vehicle health state for use by the flight crew, ground crew, and maintenance depot*"(Srivastava, 2006).

Through properly developed and validated tools such as diagnosis, prognosis and detection, the IVHM can mitigate the adverse effects during the mission. Either system/sub system or component faults during the flight act as the adverse effects. These faults may be caused due to hardware failure, degraded performance or any environmental factors or hazards. These are the goals of IVHM in a report by NASA.

According to (Srivastava, 2006), the objective of IVHM is to diagnose failure, malfunction and degraded performance, determine accurate prognosis and predict vehicle safety as well mitigate failures, damage and degradation in real-time. The word diagnosis means the identification of a fault in the system by the process of elimination using dependency or cause-effect relationship between the components. "*The prognosis reasoning would enable the IVHM system to account for deterioration in performance and/or expected useful life*".

1.2.2 IVHM Functions

Integrated Vehicle Health management concept is the capability to make appropriate decisions about maintenance actions through system level research, based on real time monitoring/detection, diagnosis, prognosis and mitigation of faults.

The function of the real-time monitoring/detection is to develop validated technologies to detect anomalies from adverse events throughout the aircraft in hardware and in software, and the interaction between the two classes of systems (Hobbs, 2009). The real-time monitoring capability is based on the advanced sensor techniques which are used to systematically track and acquire the foundational data regarding the relevant health condition. Monitoring is an essential component to the health management system since the information from it is the base of any subsequent diagnostics and prognostics.

The diagnostic capability is the action or process of identifying and determining the status of the component, or a system, or an aircraft to perform its functions based on observed parameters or through the relevant evaluation methods. In other words, diagnostics is a fault identification based on automated detection and judgement logic. Using diagnosis can find the sources of any failure or fault related to the line replaceable unit (LRU).

The prognosis capability is a specific process of predictive diagnostics which includes either the prediction of the remaining useful life or determination of the time span of appropriate operation of a component, system, or aircraft. Prognosis is a special action for operators to forecast or predict the relevant conditions before any failures occur. With prognostics, precursors can be identified and the future life of aircraft or systems will be provided for operators.

The Health Management system still has some secondary operations such as mitigation of faults. The purpose of the mitigation element is to develop onboard mitigation technologies to minimise the impact of adverse effects to ensure continued safe flight and/or landing of the aircraft(Industry Canada, 2004).

Health management processes given by (Avionics Magazine, 2011) include the following:

- Fault detection and isolation philosophy;
- Optimal sensor quantity and placement guidelines;
- Standard built-in-test designs and practices;
- Metrics, e.g., fault coverage percentage or fault isolation accuracy percentage;
- Verification and validation of plans and procedures;
- Fault modelling guidelines;
- Interface standards between subsystems and central maintenance systems.

The goal of the integrity assurance element is to develop advanced integrated assurance tools, test beds, and technologies for assessing the performance, robustness, and other integrated assurance needs(Industry Canada, 2004). The relationship between the above IVHM functions is illustrated in Fig. 1-1.

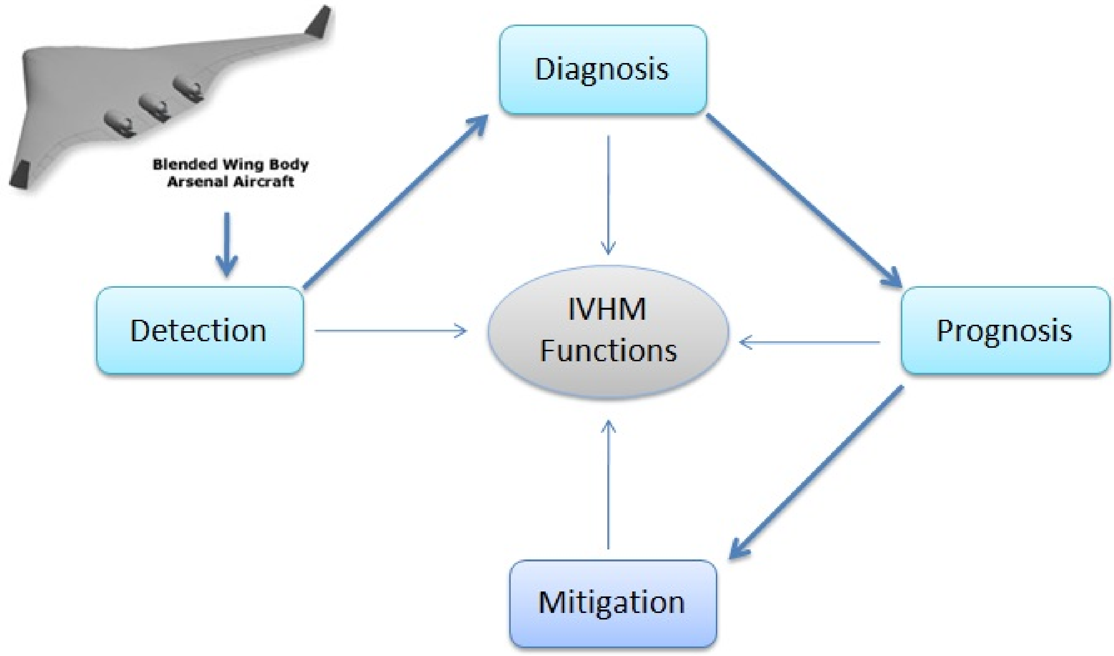


Figure 1-1 IVHM Main Functions

1.3 Problem Statement

Although Unmanned Aerial Vehicle (UAV) has become much more complex in the last 50 years, much of the tools for ensuring serviceability have remained essentially unchanged(List Lab, 2011). Simultaneously, the competition in the current aviation market is also very heated. Therefore, a new UAV with IVHM Technology will have a competitive capability in the future aviation market.

Furthermore, the IVHM technology is new to the development of aircraft fuel systems and little is known about its true functions and capabilities. Moreover, the hardware faults and failures are very hard to detect, diagnose and mitigate in-flight with existing technologies. Consequently, when these problems occur they can lead to catastrophic accidents(Vachtsevanos, 2006). This is the great challenge. Hence this thesis offers a design phase within its research to resolve these problems. Introducing IVHM technology into the fuel system development will not only improve its reliability and availability, but also benefit the overall aircraft performance. This thesis will provide a testability model of the fuel system test rig as well as fault detection and isolation on that testability model.

1.4 Research Methodology

The approach commonly referred to as Fault Detection and Isolation (FDI), or in a broader sense Integrated Vehicle Health Management (IVHM), appears to have good potential in terms of improving human safety, monetary losses, and overall mission success capability. For example: the petrochemical industry alone incurs an estimated $20 billion in losses every year due to process failure, and the cost is much higher when other industries such as pharmaceutical, speciality chemicals, and power are included.

In order to investigate the potential of FDI and IVHM schemes, Engineers at Boeing and Cranfield University are working together to develop a realistic test rig on which various techniques can be implemented and evaluated. The target system is lab-scale Unmanned Aerial Vehicle fuel system simulator. Phase 1 of the project involves the development and commissioning of the rig and the development of simulation models for the rig (which can later be used in FDI design). This thesis focuses on the schematic dependency model development of the fuel system test rig and also establishes the testability of this model using the eXpress diagnostic software.

1.5 Research Objectives

The goal of this thesis is to focus on developing fault detection and isolation statistics of a model which are suitable for the UAV fuel system test rig. The objectives of this research are as follows:

a) To meet the requirements of the Individual Research Project – IVHM on UAV fuel system test rig. The research on the diagnostic design phase will help the full scope diagnostics and prognostics of UAV fuel system.

b) To discuss the dependency model based detection/isolation statistics, advanced FMECA, and diagnostics capability of a fuel system.

1.6 Research Values

- Low risk, High-Confidence Analysis;
- Large Improvement in Safety;
- Improved Operational Integrity of the fuel system;
- Rapid maintenance turnaround time;
- Low-Cost optimisation of operational integrity.

1.7 Thesis Organisation

The project is organised as follows, Chapter 1 describes the introduction of the project, problems, research objective and methodology. Chapter 2 follows with the Literature review and chapter 3 explains the fuel rig and dependency model development. Chapter 4 describes the diagnostic study of the model and its reports. Chapter 5 discusses the model testability as well as its usability in the test rig. Chapter 6 describes further work and finishes with the conclusion.

2 Literature Review

This chapter explains the literature study carried out in order to understand the subject eXpress software, IVHM design using eXpress, accessing it and dependency modelling techniques.

2.1 eXpress Software

The eXpress diagnostic modelling and analysis tool created by DSI International provides a robust foundation for the assessment and optimisation of diagnostics design. A major feature of eXpress is its ability to support a top down modelling process which requires definition and the development of an initial top level functional design. eXpress greatly facilitates effective system testability for the complex modern systems. Design for test and design for diagnostics are the two distinct design practices which are usually referred to by the term testability.

The use of good design practices which facilitate testing is called “Design for Test”. Optimisation of a design is accompanied by test procedures which facilitate good diagnostics and refers to design for diagnostics. Testability involves the assessment of the fault detection and fault isolation capability of a system or device, as well as the optimisation of test point placement, functional partitioning and diagnostic strategies which are needed in order to meet a system’s testability requirements.

In accordance with the operational systems remediation and maintenance support needs, eXpress ensures the effective sensing is in place and the functional partitioning is optimised. The modelling technique used in the design process is easily captured and integrated into external reasoners (for both run-time diagnostics and prognostics) in order to effectively drive the operational health management and support environment. Integrated Failure Mode Effects and Criticality Analysis (FMECA) is a significant feature of eXpress. An eXpress FMECA can also be linked to a diagnostic study and ensures that each failure mode in the FMECA is detected by the specific diagnostic strategy.

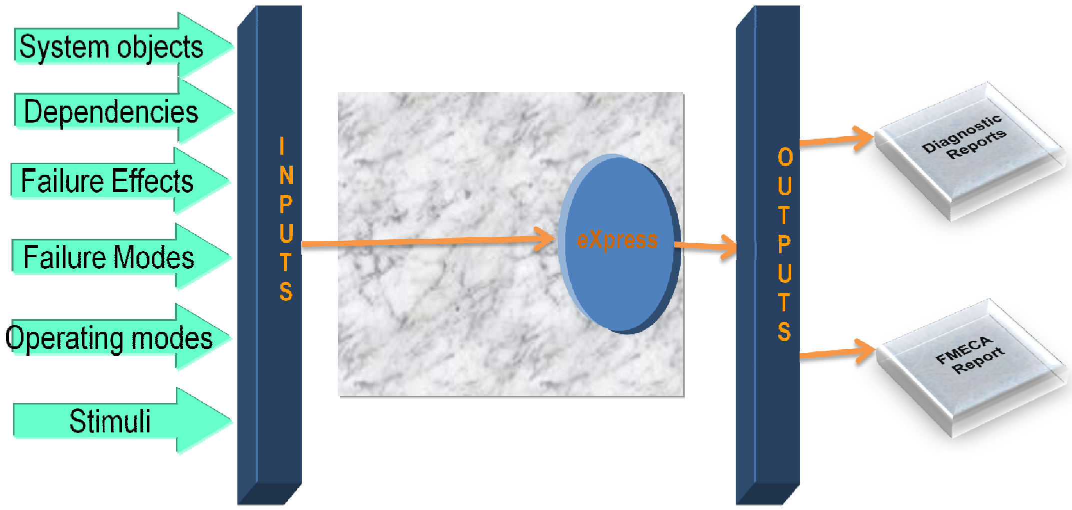


Figure 2-1Inputs and Outputs in eXpress Software

Fig. 2-1 depicts the inputs and outputs in eXpress software. In this, the inputs list from system objects such as tank, pump, filter etc..., are elements where failure modes are inserted to create fault in the component. Failure effects are the functions affected by the failure mode. The functions are passed through the element called net which carry the dependency, which connects one object into another. The operating mode is the one which gives the functions to follow through the model. All the above factors act as the input of the eXpress software and the software crate two modes of results through this they are diagnostic report and FMECA report. Diagnosis report is used to find the fault detection and Isolation statistics of the model and the advanced FMECA helps in the design process to find the effects of failure functions in the system.

2.1.1 IVHM Design Using eXpress

IVHM is a system wide coordinated approach towards fault reporting, diagnosis and remediation. It has its main roots in IVHM but quickly steps into the maintenance world as well. The benefits which IVHM bring to a wide range of problems mean that the IVHM is at the forefront of all efforts. The present challenge to the design team is in proving IVHM's merit to ensure that this brings the maximum benefit.

Fig. 2-2 shown below explains the typical solution to handling both safety critical items that often require immediate reaction, as well as important, but non real-time functions, such as trending analysis. The vehicle management functions are broader functions which factor in the mode of operation, while the area manager functions handle objects like engine management, where a great deal of importance is placed on time. In a systems engineering process, in order to support decision making this type of architecture decision must be accessed fairly quickly.

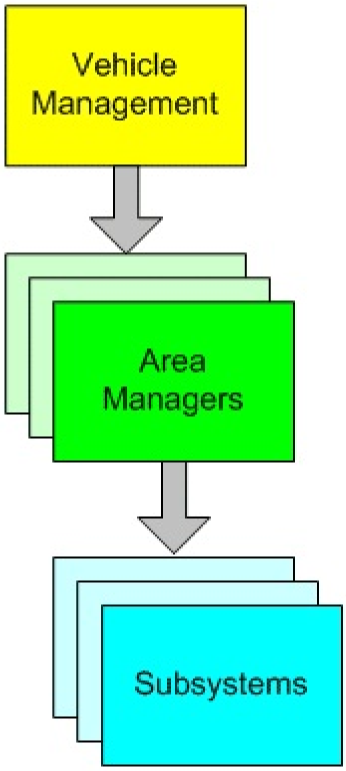


Figure 2-2 Safety Critical Items(Testability, 2011)

eXpress software is capable to quickly determine the impacts on a system diagnosis model such as detection rates, false removals, support costs, etc.

2.1.2 Accessing IVHM in eXpress

eXpress is used to provide the following statistics:

- Fault Detection and Isolation;
- Test Point Utilisation and Recommendation;
- Fault Group Size and Expected False Removal Statistics.

As processing, we can face more complex statistics which are used as main factor in modes of operation, including degraded modes of operation as follows:

- Subset FD/FI Report;
- Scope-limited Diagnostic Assessment;
- FMECA report, including Detection Method.

eXpress has a special feature called data layering with this it can influence decision making very effectively in system development. The foundation supported by its data layering which begins at the core followed by the topological model, and then continues to the outer edges by which assessment takes place. In some cases data layering in eXpress carry the features beyond a simple layering and uses a special kind of abstraction technique by which the changes to test and diagnostics approaches are simple and effective. This technique allows assessment and optimization to occur in entire design process. In order to overcome the difficulties faced in early design we should make some sort of crucial change in the design. For example

1. Earlier the functions are slowly augmented with failure modes.

2. Each functions form the connectivity between them and flow in the model.

3. When the design matures, the subsystems comes into picture with increasing amounts of detail, then failure modes become common while testing is carried.

To modify the functions and failure modes without diverging from the original approach, express provides the hybrid modelling capability, which allows the transition take place at testing level, without any loss of resolution or changes in the model. When the failure modes are entered into the system, tests is conducted component-by-component basis from functions to failure modes.

2.2 Prognostics and Diagnostics

Diagnostics and prognostics both are concerned with health assessment; henceforth we can study them together. Moreover, the decision-making roles of the two are different. Diagnosis results widely used to corrective (repair/replacement) actions; prognosis results are used forecasting which gives the way for preventive and/or evasive actions (CBM, mission reconfiguration, etc.) with objective of maximizing the service life of replaceable/serviceable components and minimizing operational risk. In many situations diagnosis and prognosis aid each other.

Diagnostic techniques can be used to identify the faults that have occurred in system performance; the motivation of prognostic methods to estimate when the faults progress to critical stage for causing system failure. Prognostics can be used to update the failure rates (reliabilities) of the system components, and in the event of a failure these updated reliability values can be used to isolate the failed component(s) via a more efficient troubleshooting sequence.

2.2.1 Diagnostics

Diagnosis is the term applied for identification of the reason for the problem. The diagnosis can be done effectively depending on application and level of maintenance activity. Diagnosis involves identification of faulty component, a failure mode, or a failure condition. Basically it finds the one or more symptoms which causing the problem. The symptoms may be within the symptoms which prevent the system to act abnormally leads to failure. The primary objective and job function of the user of diagnostic results have to determine whether what kind of activity they going to posses like address the external root cause, to address the damaged system component else both. In many large system instrumented within built sensors accompanying with diagnostic tests, these detect the failures and processing them to eliminate the failure caused in the system. In some of the applications, the failure detection and isolation are done in single step. For example diagnostic problems can be carried as classification problems, where it is associated with system corresponding either normal or failure modes. These kind of approaches are effective when the maintenance is done properly and relationship between failure modes and component is strongly implied(Liang, 2009).

2.2.2 Prognosis

As stated earlier in this thesis that prognosis (forecasting) is prediction of the outcome and probability is determine useful life of crucial document. In order to make this research objective effectively some approaches of prognosis is applied like

1. statistical reliability approach
2. trend-based evolutionary approach
3. artificial neural network, and
4. Static estimator based approach. These are developed to track and analyse components of the life(Liang, 2009).

2.3 Dependency Modelling

The important step in the formulation of diagnostic/prognostic inference process is modelling the related physical system to the observed data, for doing this there are several methods and approaches are in both research and development of diagnosis and health management. Some of model is prescribed below:

Physical Model - Physical models are used to design the system or maintaining the system operation founded in natural laws, e.g., structural mechanics (properties of materials – solid, liquid and gas), statics and dynamics of rigid bodies (e.g., finite-element models), thermodynamics, etc. usually physical models are designed in a such a way that they explain the normal behaviour of the engineering system, not the failure behaviour. The failure space of the system takes large memory than the normal engineering system. Physics-based failure models need to be specially built which required aloft of experts (scientists and engineers) manpower. Hence designing the failure modelling is expensive.

Reliability models - In order to increase the reliability of the individual components the evaluation is done using reliability block diagrams. The analysis of reliability is simply based on the probability of failure system done by empirical and laboratory data.

Probabilistic and graph-theoretic techniques are used to analyze the overall system reliability using reliability block diagrams. For the probabilistic independence of individual failure and sympathetic failure judicious assumption need to be maintained. The reliability model are utilized in identifying parts of system in need of health monitoring and diagnostics/prognostics .Component reliabilities had been used to update the data's of periodic maintenance and inspection schedules, and computed reliabilities of each sub assembly; module, etc. These can be used to eliminate the causes of anomalies or failures as effective as possible.

Machine learning models – With sufficient relevant training history this data dependent models are very efficient. Neural network is one of the prominent techniques in this class. The neural network learning demonstration is often impressive too. In a lower level the causes are connected with physical components, effects with failure of components, diagnostic tests or symptoms and the relations between causes and effects with links physically, between components or directions of energy flow.(Gould, 2004)

Dependency modelling – Based upon the need for a more rigorous and formal method of developing diagnosis, dependency modelling is developed during the period of 1950's. It is evaluated as a diagnostic technique in 1970's. The relationship between a designs testable events and functions responsible for the events are represented by the dependency model. In later stages tests are mapped to specific failure modes rather than functions fills the gap between FMECA analysis and the run time diagnostics. (STAT User's Group, 1994)

To improve a proposed diagnostic design, functional dependency models are created in the early development phase. Once implementation details are available to predict diagnostic performance and to document the diagnostic strategies the functional dependency models will convert into failure based models.(Gould, 2004).

2.3.1 Hybrid Diagnostic Modelling (HDM)

In late 1990,s, DSI International develops hybrid diagnostic modelling techniques to address both functional and failure based term within a single diagnostic model. Now in eXpress software all the capabilities are available and it is the first modelling tool to feature between functions/failure modes, HDM also represent tests used during diagnostics. The definition of failure mode details name of the failure mode, failure mode associated with percentage of component failure rate, failure mode impacted functions and affected functions relationship with failure mode. Failure mode typically affects the set of functions or sometimes affects the set of functions. Once the detailed Information about the physics of failure is not available, the possibility of sometimes affects the set of functions such as for a black box or a commercial off the shelf (COTS) device for which Built in Test (BIT) coverage percentages are provided.(Gould, 2004).

Tests are defined in terms of functions, failure modes or a combination of two. It is very much useful in developing hierarchical system designs. Once implementation data are available and when the design matures, failure modes are added to models at the lower level designs and tests defined in terms of these failure modes inherited into higher design levels. A full functional description of the system with oriented BIT test definitions provide a way to find the functional areas of the system that remain untested and also helps in adding tests.

2.4 Fuel System

The Primary purpose of a UAV fuel system is to store fuel and provide a reliable flow of fuel at a required rate. It also maintains proper fuel pressure established for proper engine and APU functioning under each likely condition. It includes any flight for which mission or certification is requested during which the engine and Auxiliary Power Unit (APU) are permitted to be in operation. Without this motive fuel flow delivered by the fuel system, the flight mission of the aircraft is unable to be sustained and inevitably comes to an abrupt stop. Therefore the fuel system is an essential element in a complete suite of systems required to assure safe flight(Ian Moir, 2008). In other words, an aircraft fuel system has a greater effect on aircraft performance and reliability than any other airframe system. As a result, utilising real-time detection and continuous monitoring of the performance of the fuel system becomes more crucial and necessary during the life cycle of an aircraft.

2.4.1 Fuel System and IVHM

It is very important to detect and acquire the operating status of UAV fuel system. Therefore, the IVHM technology is introduced into the development of aircraft fuel systems. The IVHM system ensures the accomplishment of real-time acquisition, processing and transmission of UAV fuel system's operating status and health data. As a consequence, the development of IVHM in UAV fuel system is able to further improve fuel system reliability and prevent unexpected failure. Although health management concepts have been discussed for a few years, IVHM technology is still a brand new concept for UAV fuel system development. Furthermore, it may provide a new design philosophy and synthesis for fuel system design.

3 Fuel Rig and Dependency Model Development

In this chapter, the topics covered are fuel system test rig and its schematic diagram. Also covered is the dependency model development from initial component creation to final Input/output flag insertion including fault insertion into the model and test set creation. Each and every phase of the model is clearly explained and for easy understanding diagrams are presented with the explanation.

3.1 Fuel System Test Rig

Fig. 3-1 depicts the photograph of UAV fuel system test rig in the Integrated Vehicle Health Management (IVHM) laboratory in Cranfield. It consists of a number of tanks, pumps, flow meters, pressure sensors and other instruments. This rig permits the author to implement and investigate the rig with a wide range of fault diagnostic tools and techniques. A wide range of faults of various types are injected into the test bed. Various fault detection and isolation techniques are carried out to find the fault in the test rig. The test rig also has the capability to reconfigure the components and instruments.

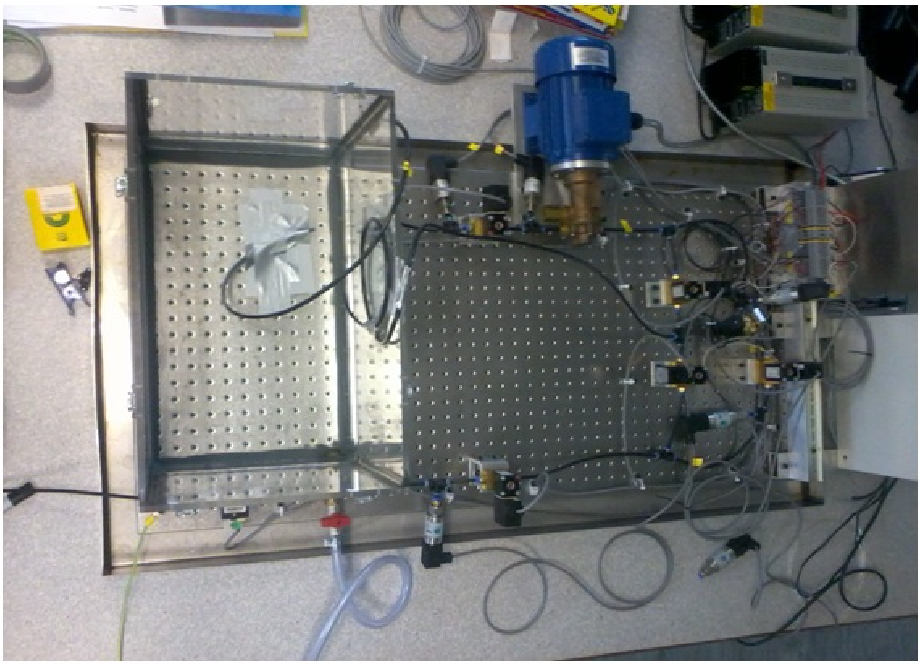


Figure 3-1 IVHM Fuel System Test Rig

3.1.1 Fuel Rig Schematic Diagram

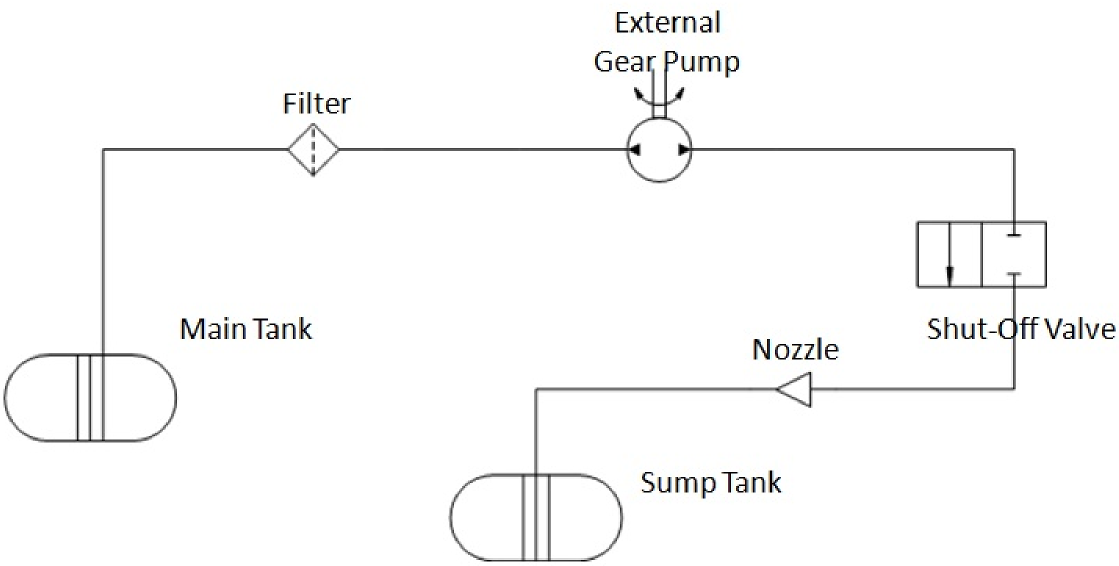


Figure 3-2 Schematic Diagram of the Fuel System Test Rig

The Schematic diagram of the fuel system test rig consists of Main Tank, Filter, Fuel Pump, Shut off Valve, Nozzle and Sump Tank. Fig. 3-2 shows the basic layout of the fuel system test rig. The Main Tank supplies fuel to the pump through a filter which is represented in the diagram. Followed by filter is the pump which pumps the fuel flow to a certain constant rate and is adjusted by the shutoff valve which then transfers the fuel through a nozzle and discharges fuel into the sump tank. In the UAV the sump tank role is played by UAV engine.

Fig. 3-3 represents the sensor connections, fault injectors and pipe carrying the fuel. Actually, the strategy of this project is to achieve maximum detectability and isolability of the schematic model without sensors and once it is successfully done; sensors are inserted into that model.

The position of the sensors especially pressure sensors as shown in the Fig. 3-1 are two across the filter, two across the valve and one after the nozzle and atlast one flow meter after the nozzle to find the overall flow.

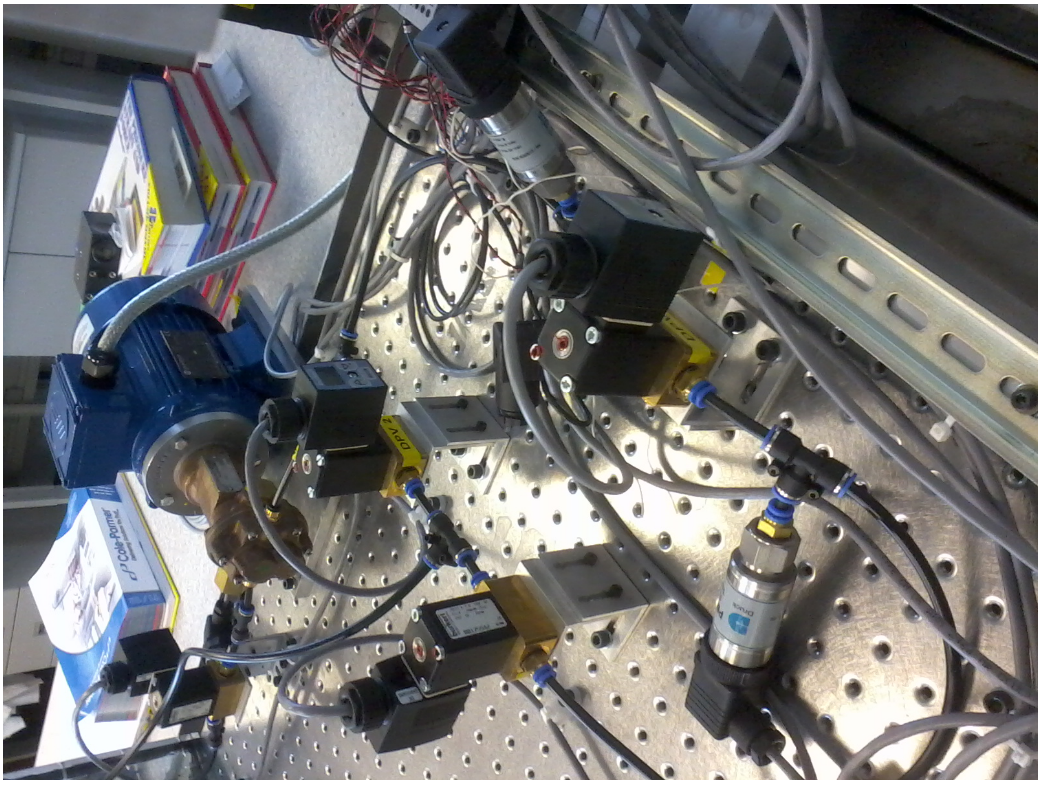


Figure 3-3 Sensors and fault injectors

3.2 Model Development Procedure

The Model development begins with the creation of a new file in the eXpress interface which is similar to that of windows for easy browsing through the design elements. The eXpress user interface screen is shown in the Fig. 3-4:

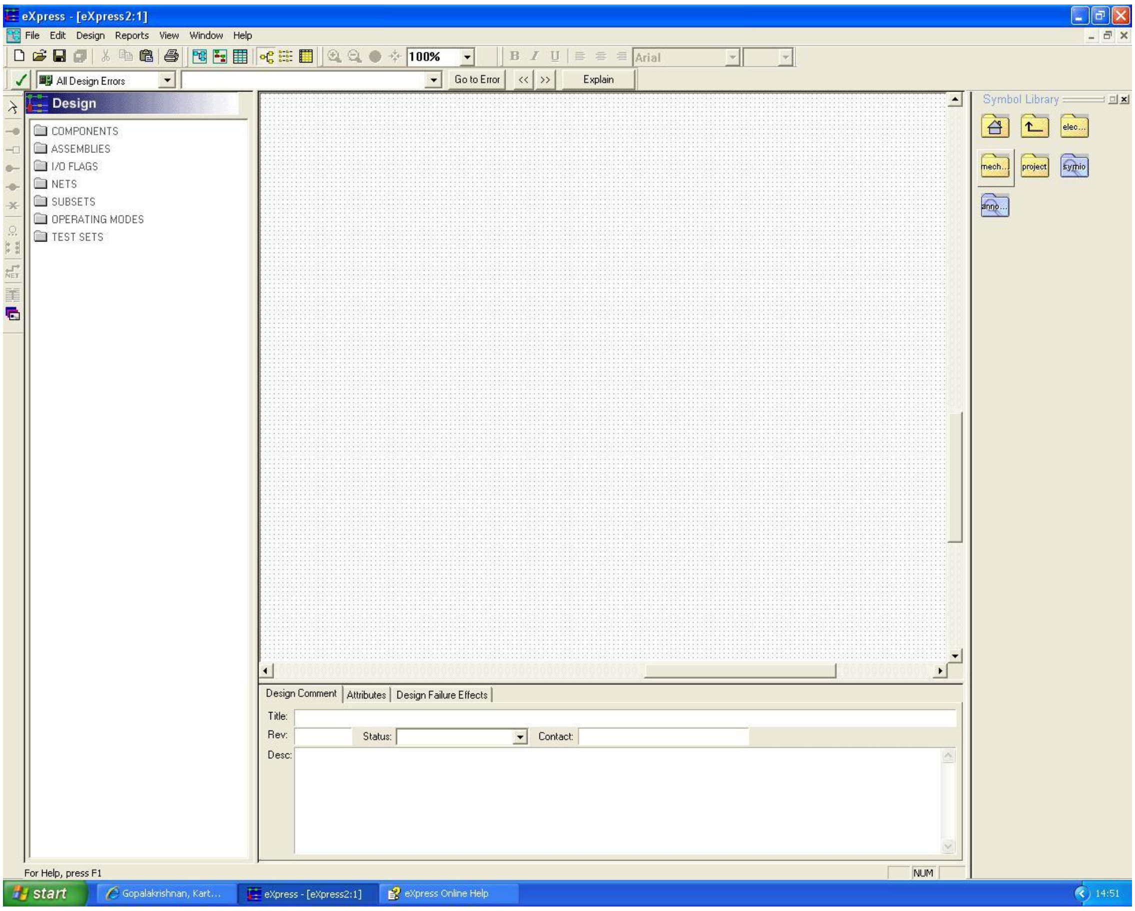


Figure 3-4 eXpress Interface Main Page

3.2.1 External Model Development

In the initial phase of model development begins with a new file created with a template from the file menu. It is possible to create a new template sticking with the existing template. For this model the basic attribute template has been selected as shown in Fig. 3-5. The file is then saved in the destination folder after giving a name for it. In the design comment panel shown in Fig. 3-6, one can document the title and description of the design, along with configuration control information.

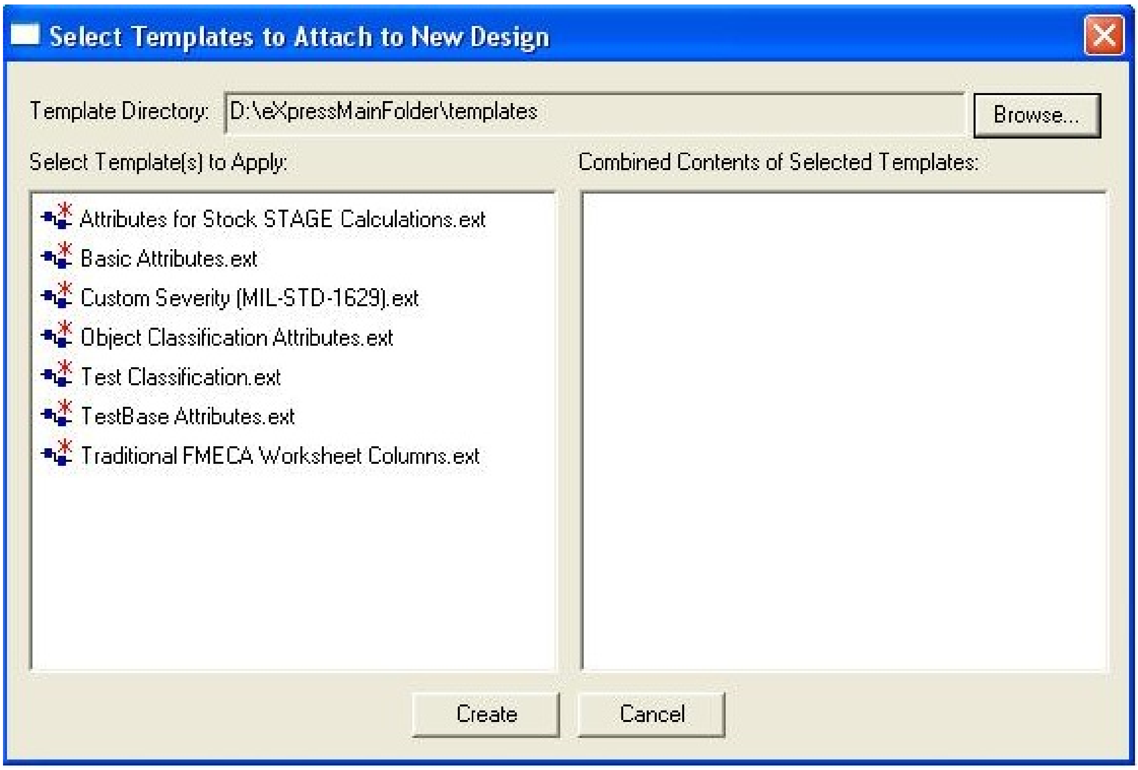


Figure 3-5 Attribute Selection Panel

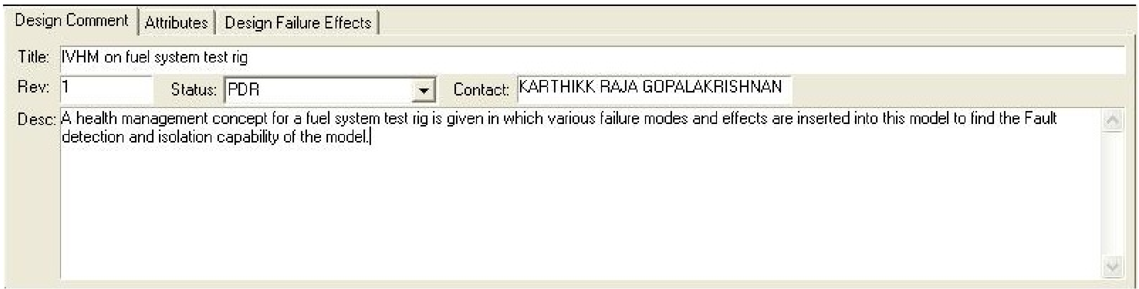


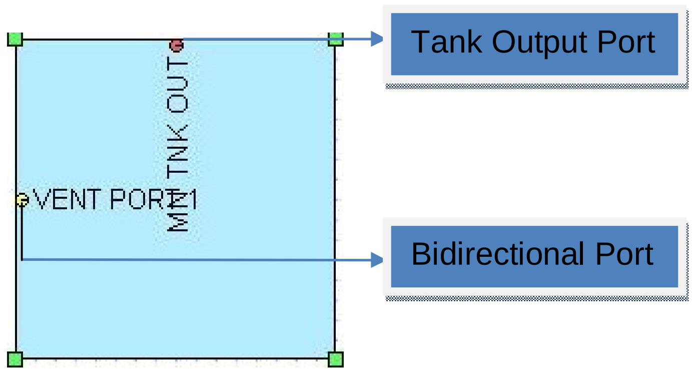
Figure 3-6 Design Comment Panel

3.2.2 Object Creation

After creating the new file, start creating objects from the standard editing mode. The symbol library panel shown in the right side of the design sheet window is used to insert components in the form of symbols inside the design sheet window. In the object specific context panel which will open when a single object is highlighted provide the options of object detail, object failure mode, object states and failure effects. In the attribute option the user defined values such as cost and time of that particular model can be viewed. In the object batch changes panel, it is possible to change object type, colour, size, alignment, orientation, distribution and it is also possible to arrange, convert, align or distribute, rotate or flip and cut, copy or paste. For this model the objects/components created are Main Tank, Filter, External Gear Pump, Shut-off Valve, Pipe, Nozzle and Sump Tank. (eXpress Quick Start Guide, 2003).

3.2.2.1 Main Tank:

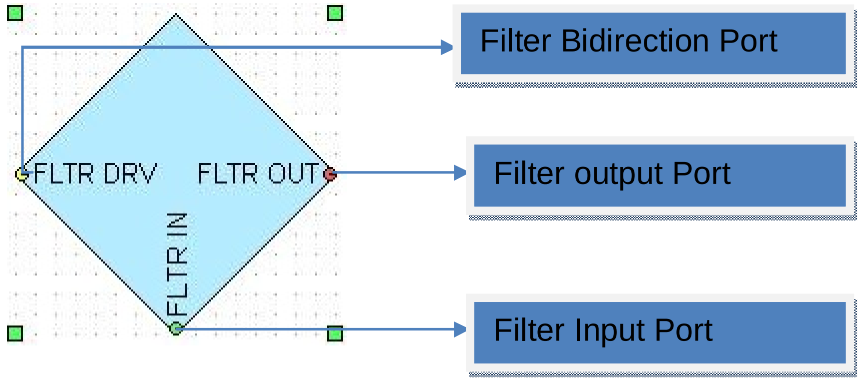
The Main Tank acts as a safe container for flammable fluids. Its primary purpose is to store fuel. These tanks vary in size and complexity from small plastic tanks to large cryogenic tanks.



- Main Tank must allow or provide the following
- Storage of fuel
- Filling
- Fuel Level Indicator
- Venting
- Feeding to the engine

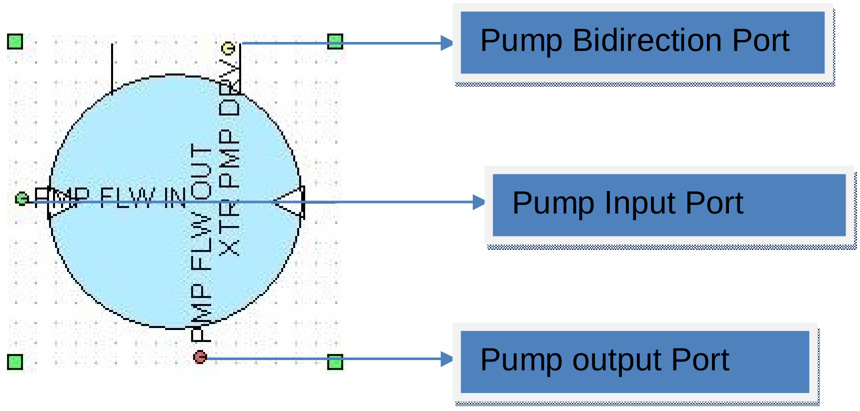
3.2.2.2 Filter

A fuel filter serves the function of a filter in a fuel line which screens out dirt and rust particles from the fuel. The filters serve as a vital function in today's modern, tight tolerance engine fuel systems. It increases performance as the fewer contaminants present in the fuel, the more efficiently it can be burnt. In some filters water drain valves drain the water which is present in the fuel.



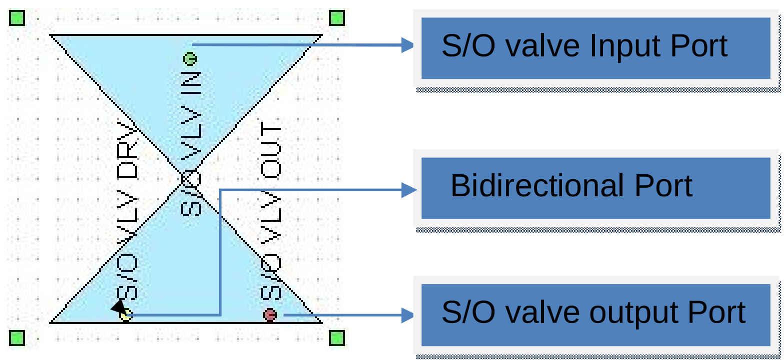
3.2.2.3 External Gear Pump

Fuel Pump is an essential component and plays an important role in non-gravity feed designs. Fuel has to be pumped from the fuel tank to the engine and delivered under low pressure to the carburettor or under high pressure to the fuel injection system.



3.2.2.4 Shut-off valve

This is a safety valve mainly used to close a line and stop the fuel flow. It is of two types, manual and automated.



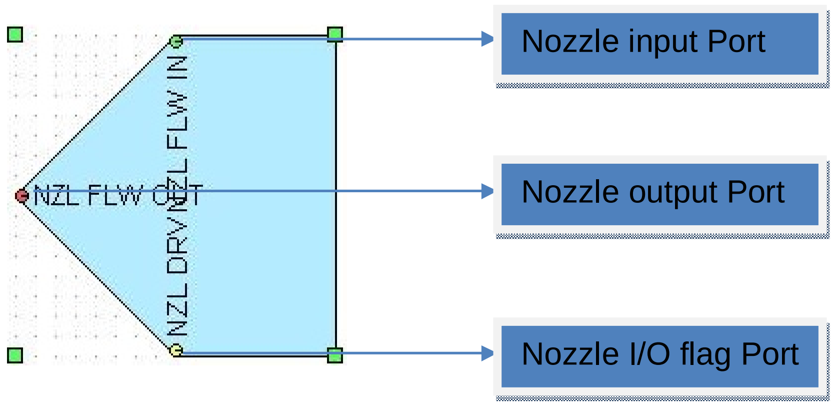
3.2.2.5 Pipe

The main purpose of this fuel pipe is to transfer fuel from one destination to the other.

3.2.2.6 Nozzle

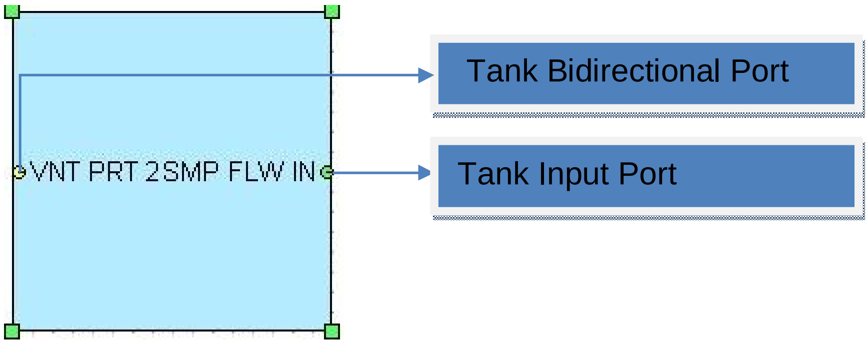
A nozzle is a device designed to control the direction or characteristics of a fluid flow (especially to increase velocity) as it exits (or enters) an enclosed chamber or pipe via an orifice.

A nozzle is often a pipe or tube of varying cross sectional area, and can be used to direct or modify the flow of a fluid (liquid or gas). Nozzles are frequently used to control the rate of flow, speed, direction, mass, shape, and/or the pressure of the stream that emerges from them.



3.2.2.7 Sump Tank

In this project the fuel system test rig has a sump tank instead of an engine which receives the fuel from the nozzle.



3.2.3 Adding Ports

Ports are added to the components once it is defined in the design sheet window. For a simple object there must be an input port and output port, based on the design needs, ports are added to the object. To add names to the ports, simply enter into the port specific context panel and create it. Fig. 3-7 shown below indicates the ports of each component used in the model.

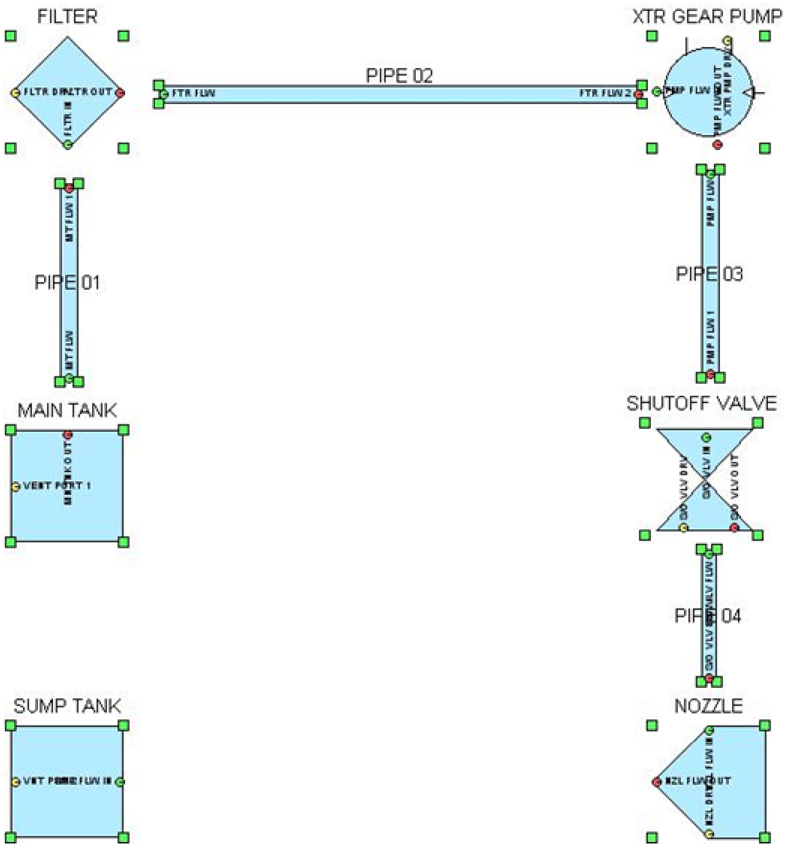


Figure 3-7 Model Component with Ports

3.2.4 Creating Nets

Now the entire model has all the necessary objects, the next step is to connect the objects through ports to carry the dependency. To connect objects via ports, net plays an important role. By entering into the net editing mode it is possible to create nets between two objects. Join the two ports from output to input and vice versa. Continue to add as many connections as needed to pass the dependency completely. The net specific context panel shown in Fig. 3-8 is used to enter the details as well as the appearance and attributes of the net. In net batch appearance we can change the colour, type (wire or pipe), and thickness of the net. (eXpress Quick Start Guide, 2003).

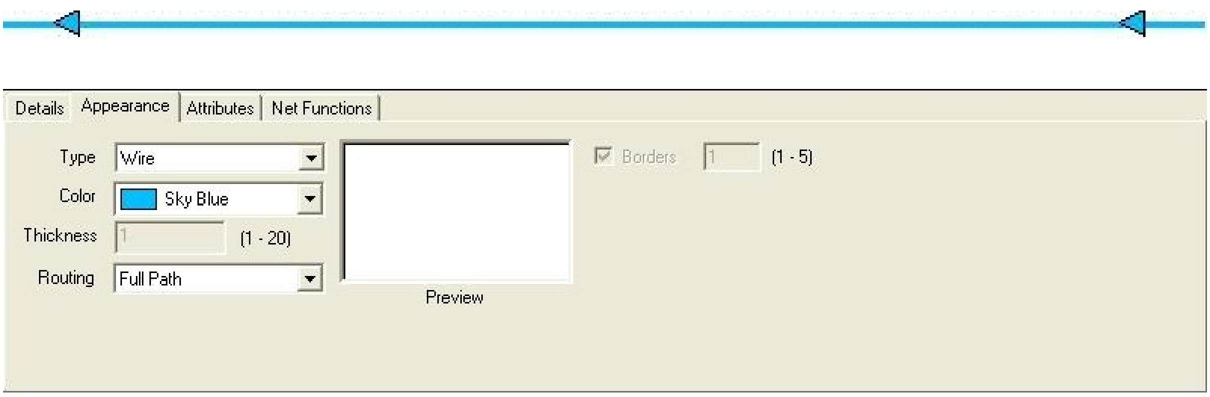
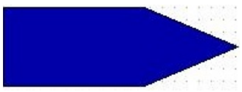


Figure 3-8 Net & Net details Panel

3.2.5 Creating I/O Flag

The next step is to add the Input/output flag as this is necessary with model elements that represent a design's interface with the outside world and with other models. It is taken from the symbol library. In the I/O object context panel the user can enter/modify the details, colour and text.



The outer design of the model is finished and it is shown in the Fig. 3-9 below with annotations to depict the heading and the border.

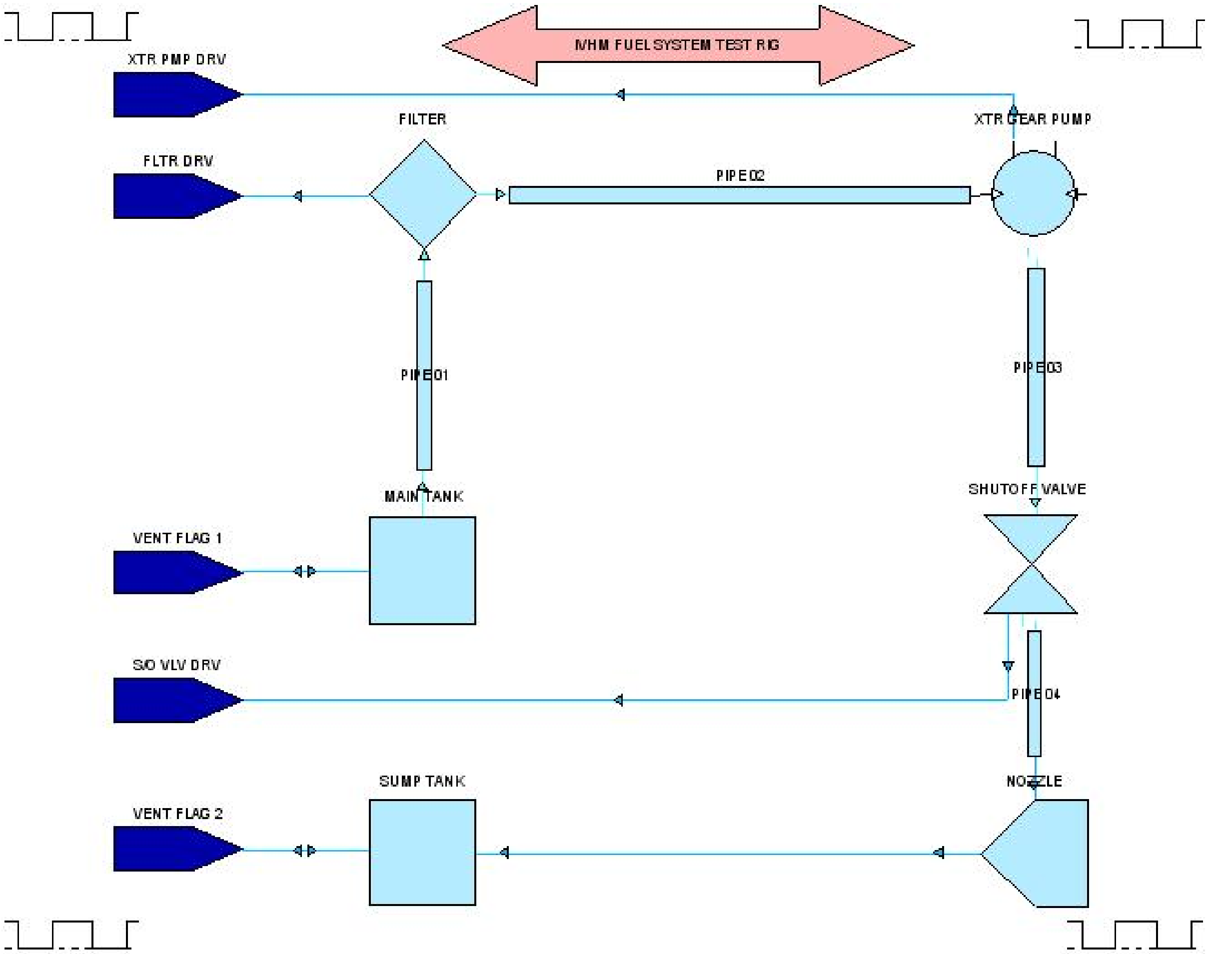


Figure 3-9 Fully Completed Model Design

3.3 Fault Insertion

Fault Insertion is the most prioritised stage in this model development. As per the current availability of the sensors and software to isolate and detect faults, failure is categorised and allocated to each individual component. More critical faults are taken into account and inserted into the model for testability.

3.3.1 Failure Mode Creation

Failure mode insertion is the first and foremost part of failure insertion into the model. To define failure mode to an object, select the object context panel as shown in Fig. 3-10 and select the failure mode. When in this mode, enter the possible failure modes of the particular component. For this model the failure modes defined for the corresponding components are clogged filter (fully/partly) for filter, non-operating or operating partly for pump, stuck open/close, sticking, (internal/external) Leaking for shut off valve, leaking (fully/partly) for pipes and finally clogging (fully/partly) for nozzle.

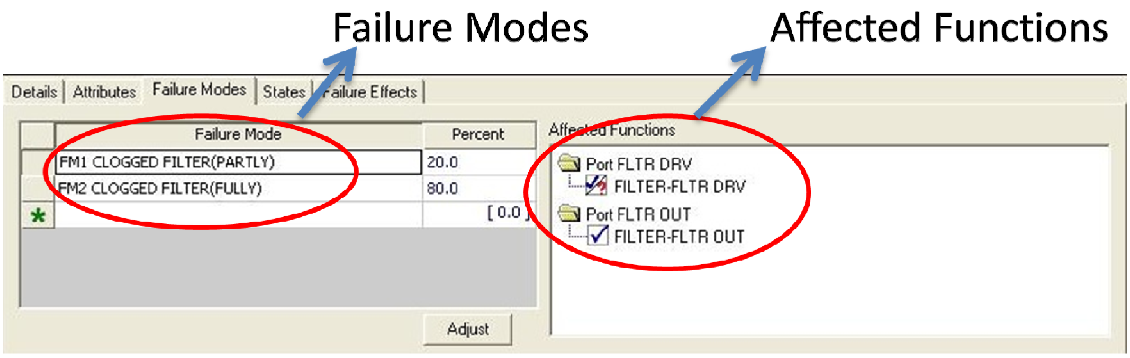


Figure 3-10 Failure mode Insertion Panel

In Fig. 3-10, the failure mode for the clogged filter (fully) is shown; where the FM2 clogged filter fully explains about the filter operation leads to contaminated fuel flow, and the probability of this failure mode occurring is about 80%. The FM1 clogged filter (partly) indicates that the filter is rusted or deposited by debris or contaminants and this reduces the performance of the filter and the probability of occurrence is 20%. The right side of the panel presents the affected functions which are affected by the particular failure mode. The user must define the affected functions for each failure mode.

3.3.2 Object States Creation

Followed by the failure mode, object states must be defined to the model in the same manner as that of failure mode creation. In this panel, as shown in Fig. 3-11, the possible functions which are active to the states are covered.

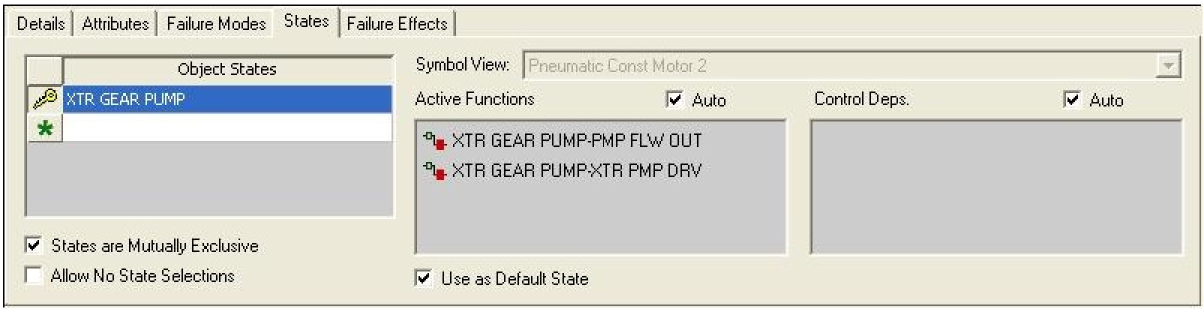


Figure 3-11 Object States creation Panel

3.3.3 Failure Effects Creation

Finally, failure effects must be defined for the functions. Failure effects defines the effect of the failure mode and object state functions. In this software there are two types of failure effects, they are object failure effect and design failure effect. Adding failure effects to the model is similar to that of failure mode and object states. Object failure effects are defined in the object failure effects panel and design failure effects are defined in the blank space within the design sheet window. The difference between the design failure effects and object failure effects are as follows. The causes of design failure effects as shown in Fig. 3-12 are object failure effects and the causes of object failure effects are failure modes and its functions. In both the panels it is possible to describe the causes of failure, severity and observation. Only loss of operation and degraded performance are defined as the causes for the failure effect as the failure impact is more concentrated on this only. (eXpress Quick Start Guide, 2003).

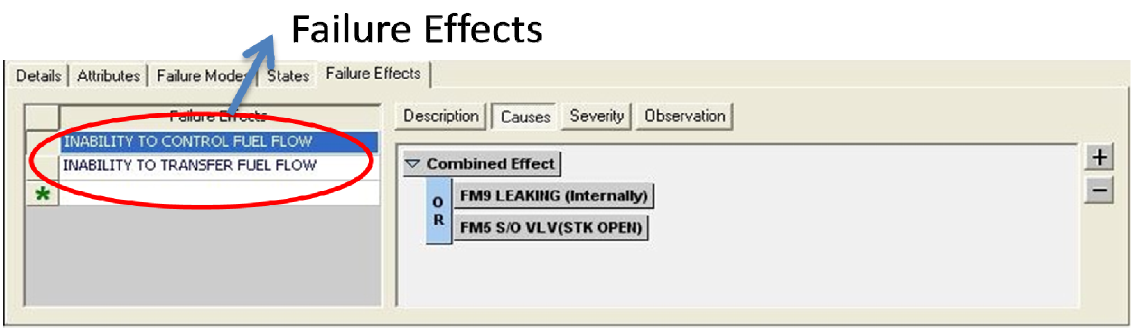


Figure 3-12 Creating Causes for the Failure Effects

3.4 Testability Development

Test plays an important role in the model development. Defining proper test into the model is a crucial role in the model development. There are eight types of test offered by eXpress software. The different test type in eXpress does not reflect different testing technologies, but rather provide different ways of determining the coverage of a test.

- Operational Test
- User-Initiated Test
- Probe Test
- Signature test
- Inspection Test
- Group Test
- Hierarchical Test
- Unknown Test Type

Each test has its own credentials and not all the tests are required for the model to find fault detection and isolation. The tests used in this model as shown in the Fig. 3-13 are Inspection Test for inspecting Filter, Operational Test for Pump and valve operation and signature test for the overall fuel flow.

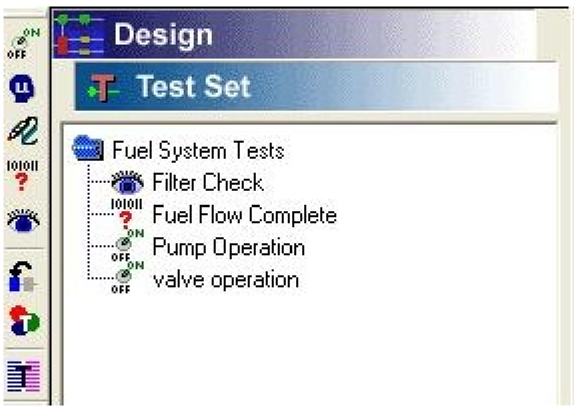


Figure 3-13 Test Set Explorer Tree

The tests are selected based on the following criteria. The Operational test is used to define a test that examines all functions or failure modes that can be observed at a given output of the design and quickly accesses the overall detectability or whether an entire design segment is operational.

For this model the test set name is defined as fuel system tests. Corresponding tests desired upon the user need of fault detection and isolation are inserted into the design by selecting it from the test set tool bar. Fig. 3-14, 3-15 and 3-16 explain the filter inspection test, pump and shut off valve operational test coverage areas.

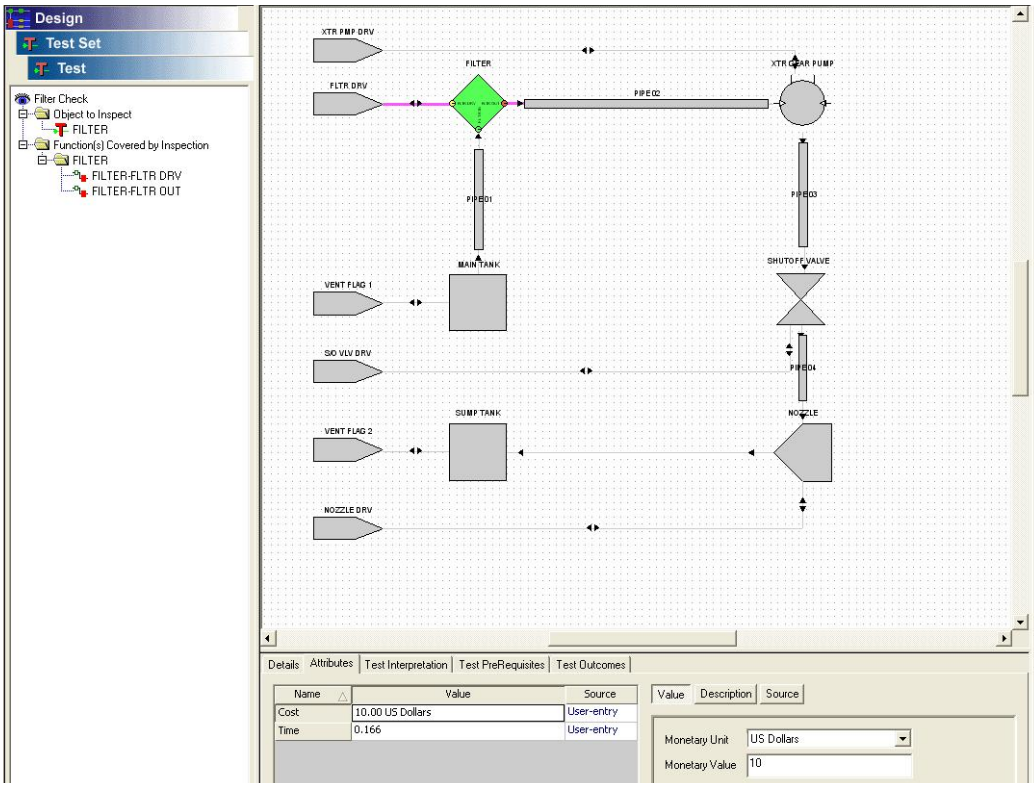


Figure 3-14 Coverage of Filter Inspection Test

Inspection tests are used when the status of one or more components can be determined under the following conditions:

- Independent of the component's role in the system (using visual inspection, external test equipment, etc.);
- Using ambient means (air temperature, sound etc.);

- Using non-topological “rules”;
- Using prognostic algorithms.

Adding Inspection test to the model is similar to that of operational test as shown in Fig. 3-14 but instead of selecting I/O flag select all the objects that contain functions to be covered by the test. The inspection test comes in three varieties which are:

- Inspection for operation/malfunction;
- Inspection for operation;
- Inspection for malfunction.

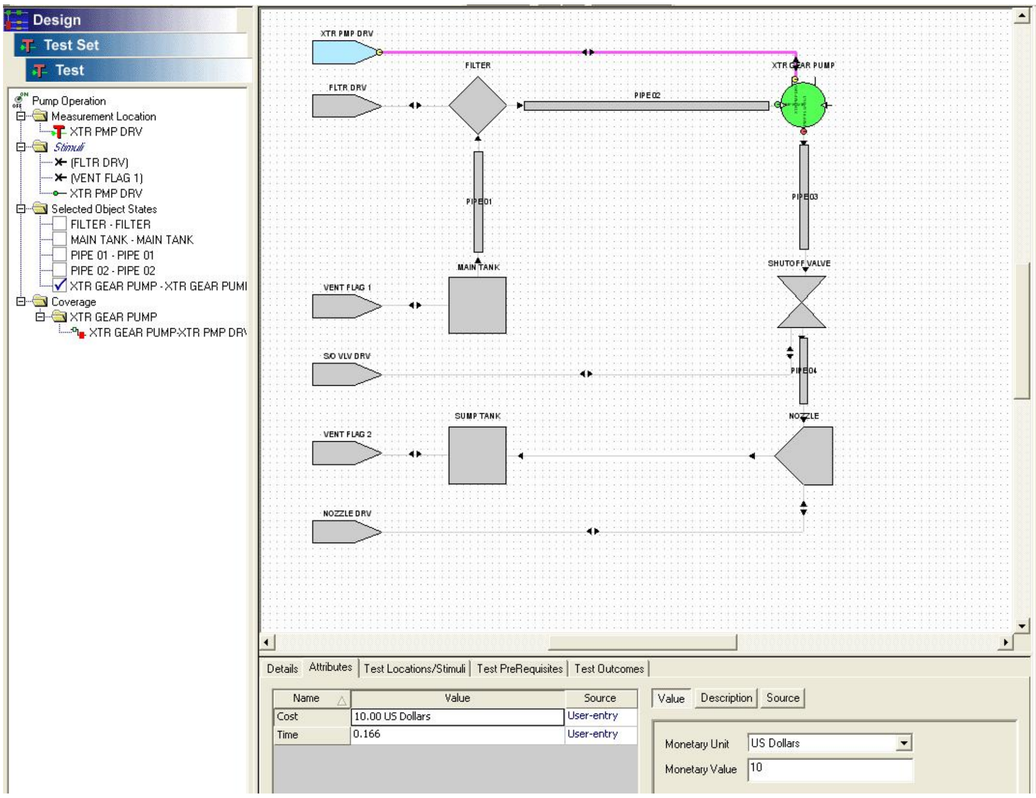


Figure 3-15 Coverage shows Pump Operational Test

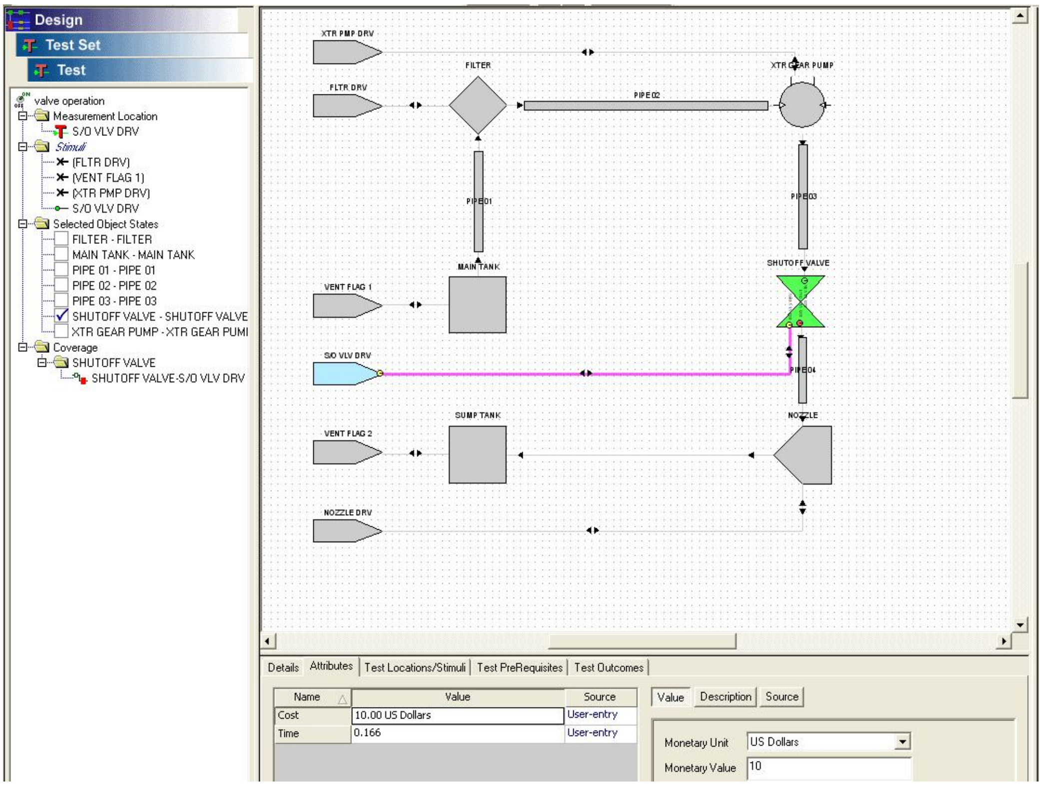


Figure 3-16 Coverage shows Shut off valve operational test

The inspection for operation/malfunction is selected for this model because in this variety the status of the covered functions or failure modes can be fully determined.

Signature tests differ from the previous types of tests in that they are defined by picking the specific functions or failure modes that are to be covered. Although signature tests are particularly good at testing individual functions of the design, a diagnostic strategy that uses only signature tests will often be grossly inefficient and computationally intense. Hence there is a need for creating tests in addition to this signature test to make the model more efficient. These are described above in the previous section. Fig. 3-17 shows the coverage area of the signature test in the model.

Adding Signature test to the model is similar to that of inspection test. Expand the “Location for testing” folder in the Explorer tree and mark the check boxes for all desired test points. Similar to the inspection test, the signature test also has three varieties and for this model signature identifies operation/malfunction is selected as the existence or non-existence of the desired signature directly indicates the goodness or badness of the covered functions.

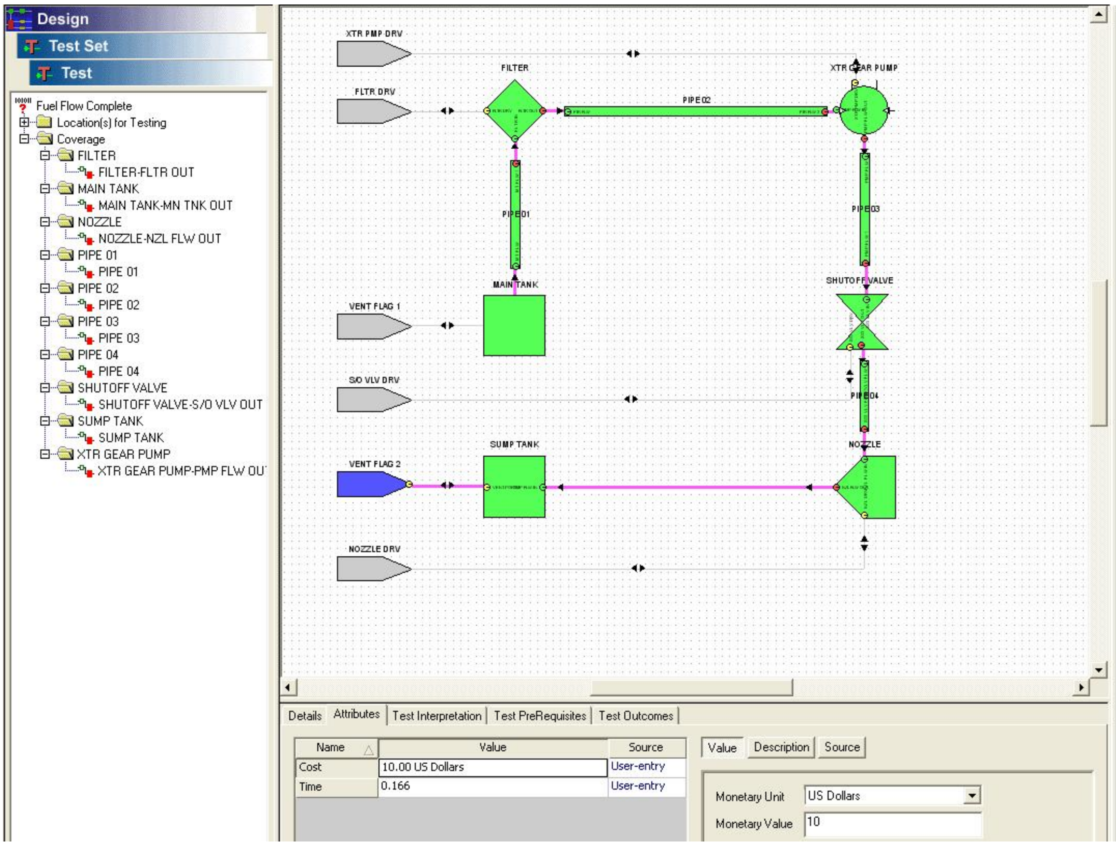


Figure 3-17 Coverage of Signature Test

3.4.1 Subset Creation

Subsets used to identify a specific set of objects, ports or failure effects that can be used to constrain diagnostics or segregate calculations within certain diagnostic reports. There are five types of subsets available in the given software eXpress, they are:

- Attribute based: with the Attribute based subsets, the analyst selects a checklist attribute which has been associated with objects which have specified values for the selected attribute;
- Combination: with the combination based subset the analyst can combine multiple subsets using the set operators add, subtract, union and intersection. An initial state can be defined for the subset (so the

subsets can be populated by subtracting other subsets from the entire design);

- Explicit selection: with the explicit selection subset the analyst explicitly populates the subset by selecting the objects, ports or failure effects that are to be included in the subset;
- Failure Severity Based: with this subset, the analysts select an operator and severity level;
- Path: With the path subset, the analyst selects a location and the direction in which the design is to be scanned. When compiled, the path subsets are populated with all ports that are up stream or down stream from the specified location. The analyst can also select whether ports on I/O flags are to be included in the subset.

The subset type used in this model is Combination and referenced subset in explicit selection. The name created for this subset is Main tank feed and Main tank feed explicit correspondingly. By selecting a new subset in the explorer tree a subset is created for the model. It is also possible to define its properties in the subset panel. Later by compiling the subset into the model, it becomes active into the model. Fig. 3-18 depicts the subset creation and reference subset creation panel.

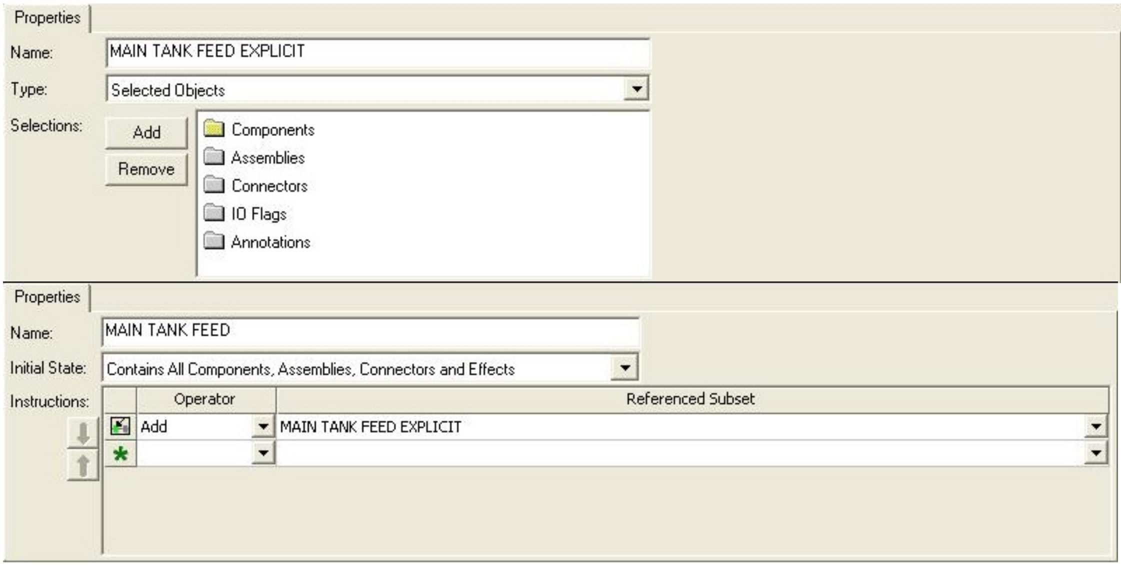


Figure 3-18 Subset and Reference Subset creation panel

3.4.2 Operating Mode Creation

Once a subset is defined, the most important thing is to define operating mode to the model. Creating this involves the same procedure to that of subset (3.4.1), but in this operating mode detail panel shown in Fig. 3-19 the subset relates to the operating mode must be defined and the description of it is also detailed.

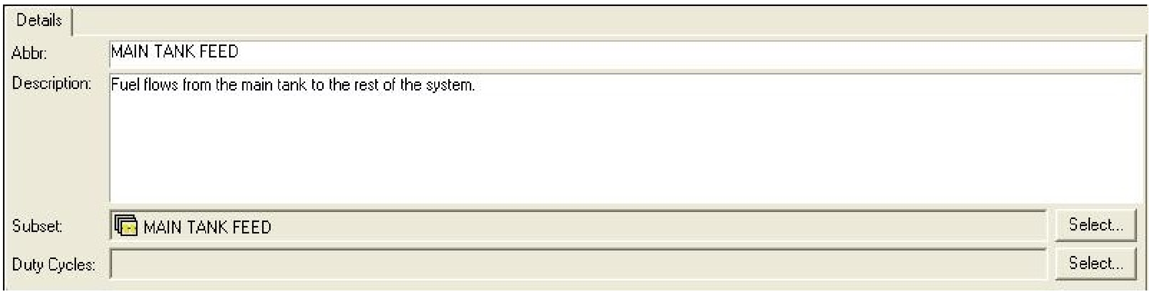


Figure 3-19 Operating Mode Creating Panel

3.4.3 Error check

Once the design is completed it is very important to check the error in the design as there are various possible errors arise during model creation. Errors are not taken into account during model creation. Once the model is finished error check the model through the error check tool bar. It is also possible to get the explanation of the error and the type of error through the error check tool bar.

The errors are sorted into four colour coded categories based on their severity. Definite (red) and Probable (magenta) errors will prevent diagnostics and FMECA studies from processing. Possible (blue) errors identify situations that will not prevent processing but should be carefully examined to determine if they are intentional or not. Warnings (yellow) represent modelling issues that should be reviewed and possibly corrected yet may have no net effect upon model analysis.

3.5 Model Assessment

Once a model has been developed, the author must verify that the model is valid and that it accurately reflects the source data (Schematics, functional block diagrams, etc...) upon which it was based. The author also verifies that all design errors have been corrected in the model. Design errors are identified using the error checker tool bar. For all complex errors the solutions are acquired through eXpress online support.

There are five types of report generated using this software and they are:

- Bill of Materials: provides list of all objects and the descriptions of this fuel system model. It is to compare the data sources to verify model completeness. The bill of material for the fuel system model is given in Appendix A.1 Bill of Materials;
- Basic Design Statistics: contains numerous quantitative model measures and provides links to numerous sub reports. These statistics are particularly useful for model validation and are the sub reports that list reference designators, tests (by interpretation), output functions and failure effects (by cause). The basic design statistics for the fuel system model is shown in Appendix A.2 Basic Design Statistics;
- Design Hierarchy: this shows how models are related hierarchically. Large system models require various levels of modelling to capture

Interoperability of lower level subsystems. Hierarchical design enhances the users' ability to represent the design of the total system both clearly and accurately;

- Feedback loops: this report identifies signals and functions within the model that participate in feedback loops;
- Test set contents – Complete set of data presenting the entire test sets within the model. With the sources used for constructing the model in eXpress, this data is used to compare

4 Diagnostic Analysis

Diagnostic study is a document within which the diagnostics of the model are generated, optimised and assessed for an eXpress design. This analysis provides the result regarding effectiveness of a specific diagnostic procedure and the overall ability of a design to support diagnostics. Diagnostic study can be modified to provide different statistical combinations and analysis based on the program needs (requirements allocation, trade off analysis, Interim design assessments, etc...). The reports generated through this diagnostic study will be useful for comparing and analysing the results, as well as modifying a baseline study.

This chapter explains generating a diagnostic study through various detection and isolation algorithms, followed by diagnostic flow diagrams and detection and isolation reports.

4.1 Creating Diagnostic Study

By using the diagnostic study context panel, the user creates the diagnostic study. In the detection options panel, the user defines the detection and isolation algorithm for the model from the long drop box which is shown in section 4.1.1. In this detection options panel the user select the test candidates such as I/O flag, Net functions and Fuel System Tests to generate diagnostic analysis based on these as shown in the Fig. 4-1and 4-2 below.

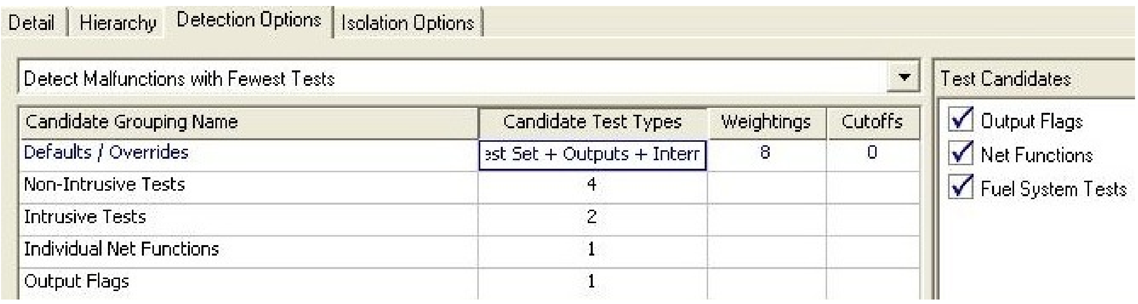


Figure 4-1 Detection Option Panel

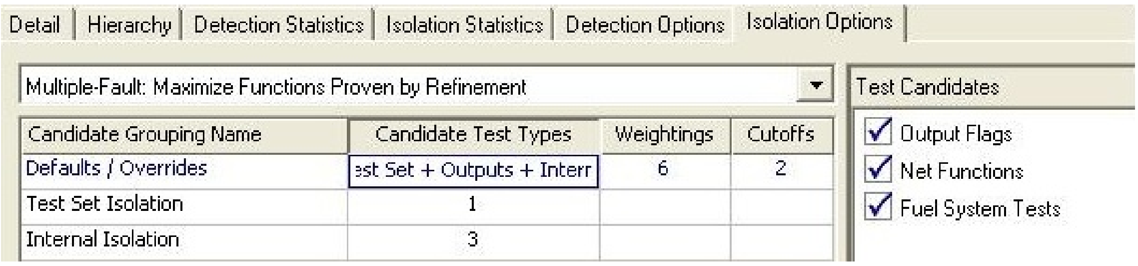


Figure 4-2 Isolation Option Panel

4.1.1 Diagnostic Algorithm

Diagnostic Algorithms are collections of settings that influence the order of test for Fault Detection or Fault Isolation. Each diagnostic algorithm is comprised of a set of Test Candidate Groupings, Test Weightings and Test Cut-offs. The groupings and weightings defined for each algorithm are the result of a sophisticated understanding of the test selection criteria that, as a rule of thumb, tend to produce "good" diagnostics in a variety of diagnostic situations (production testing, regular maintenance, trouble-shooting, damage assessment, etc.). There are seven different modes of detection and isolation algorithm that an author can select they are

- Detect Malfunction with fewest tests
- Detect probable malfunctions
- Detect critical malfunctions
- Prove operation with minimum number of test
- Before detecting malfunction, prove the maximum operation
- Minimize switches in monitored stimuli

In the isolation options panel, select isolation algorithm in the same manner as that of detection option. The description of each algorithm is explained in the 6Appendix B . The various algorithms as it contains are

- Multiple-Fault - Half split Failure Probabilities [Refinement Postponed] : This algorithm uses multiple failure isolation and does not Test set Test selection using Test Candidate Groupings

- Multiple-Fault - Half split Failure Probabilities [Refine where appropriate]
- Multiple-Fault - Half split Failure Probabilities. [No Refinement]
- Multiple-Fault - Half split Failure Probabilities [Operational Refinement not Postponed]
- Multiple-Fault: Maximise functions proven by Refinement
- Common cause - Half split failure probabilities
- Common cause - Half split failure probabilities[Max Depth =10]

4.1.2 Diagnostic Flow Diagram

The detection and isolation algorithm shown in Tab. 4-1 is only explained in the thesis as the results produced by the other algorithm are similar of this kind.

Test Option	Detection Option	Isolation Option
1	Detect Malfunction with fewest tests	Multiple-Fault: Half split Failure Probabilities [Refinement Postponed]
2	Prove maximum operation before detecting malfunction	Multiple-Fault: Maximise functions proven by Refinement

Table 4-1 Selected Detection and Isolation Option

Once algorithms are defined for the model, the user generates the diagnostic study. The diagnostic flow diagram generated based on the algorithm represents the test sequence for the calculated diagnostics. The diagnostic flow diagram is shown below in the Fig. 4-2, 4-3 & 4-4, 4-5.

Typically, the diagnostic flow diagram is comprised primarily of two types of cells: test cells and fault group cells. Test cells contain the name and type icon for each test used for detection or isolation. Detection tests are listed vertically and isolation tests are listed horizontally within the diagram. The green line that emerges from the bottom of each test cell shows the path to follow when the test passes and the red line when the test fails.

By selecting an individual test or fault group cell in the diagram, the entire test path leading up to that cell is highlighted. Cells are coloured green if the corresponding test must have passed in order to reach the highlighted cell and red if the corresponding tests must have failed. By selecting an individual cell, the cell contain only the path leading up to that cell is shown and it is possible to rebuild the cell.(eXpress Online Help, 2011).

Furthermore, by selecting cells within the diagram, the model displayed on the design sheet interactively changes colour to reflect the cumulative diagnostic conclusion (e.g. suspected and proven components) for all cells in the test path leading up to the highlighted cell. The colours displayed in the design sheet are:

- Light Green – Indicates the component proven good and it is not yet suspected;
- Green – Component is proven good;
- Light Blue – necessary for the test to insert a stimuli;
- Yellow – Component in the current suspect set is suspected;
- Teal – Component is suspected as a secondary suspect, it could be responsible for a failed test that performed previously and not in the current set;
- Dark Red – Potential Components failed;
- Light Red – Top part for the failed potential component;
- Dark Blue – Test Point for Input/output.

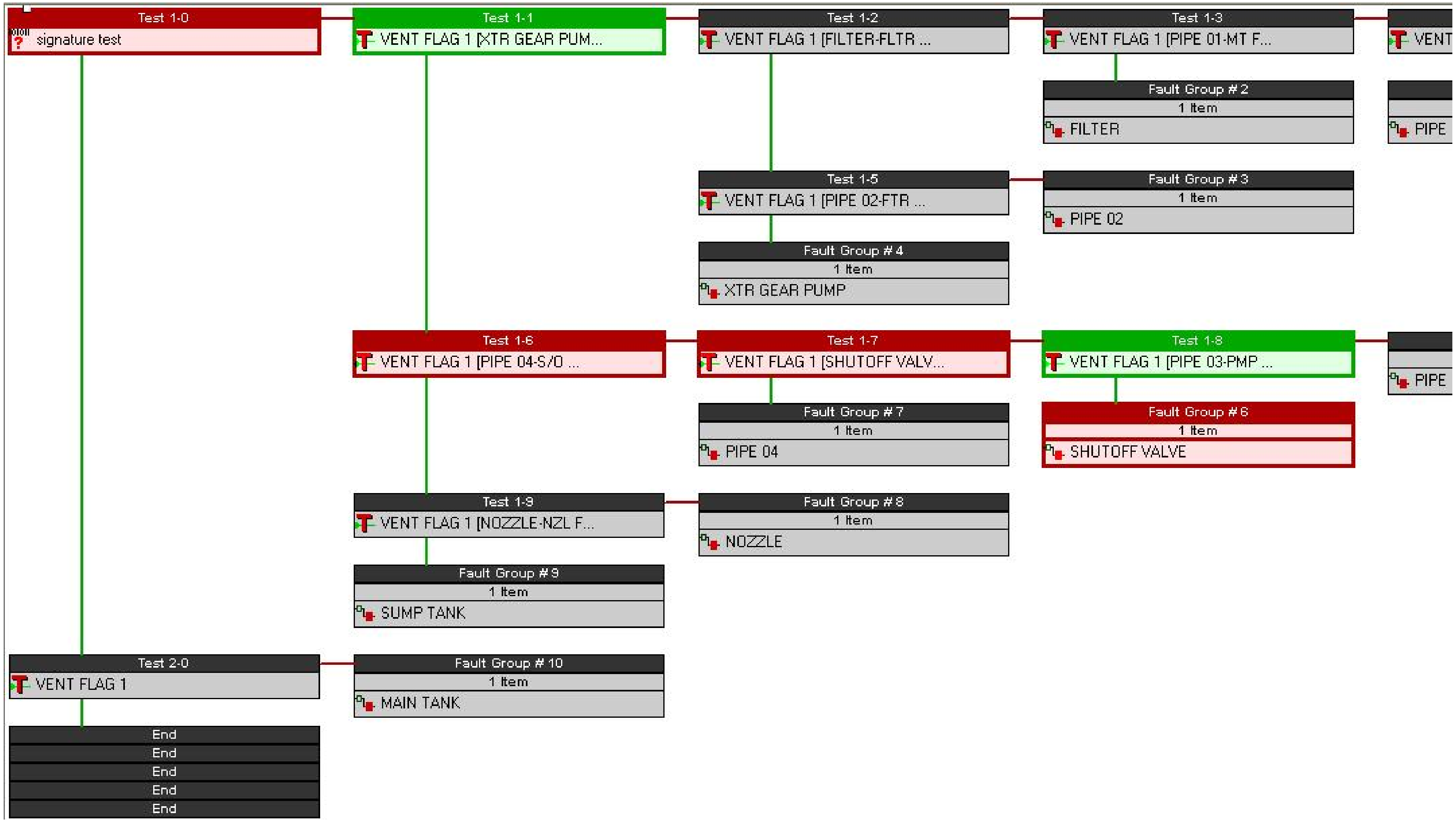


Figure 4-3 DFD with fewer test and refinement postponed

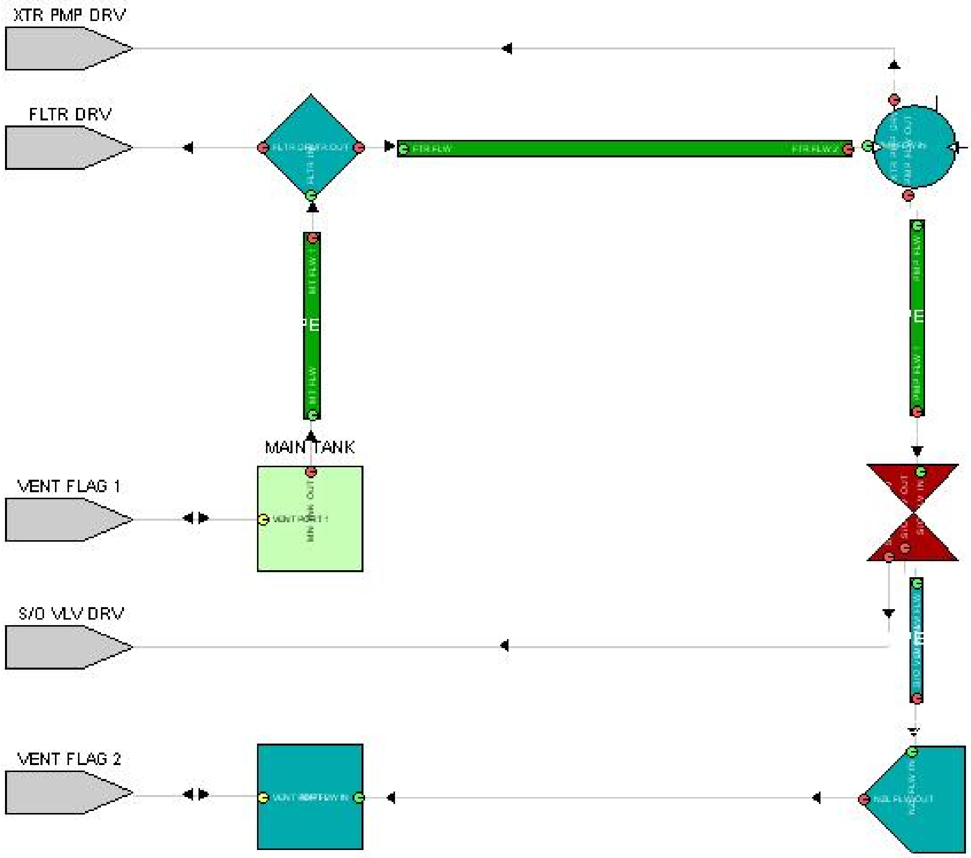


Figure 4-4 System Model with fewer test and refinement postponed

The above diagrams Fig. 4-3 and 4-4 explain the fault detection and isolation of the shut off valve through flow diagrams and through system model illustrations based on the algorithm 1 used in the model. Fig. 4-5 and 4-6 below depict the fault detection and isolation of Pipe 04 and it is shown with model illustration based on the algorithm 2 shown in Tab. 4-1. Detailed descriptions of the colours and lines indicated in the diagram are explained in the following section.

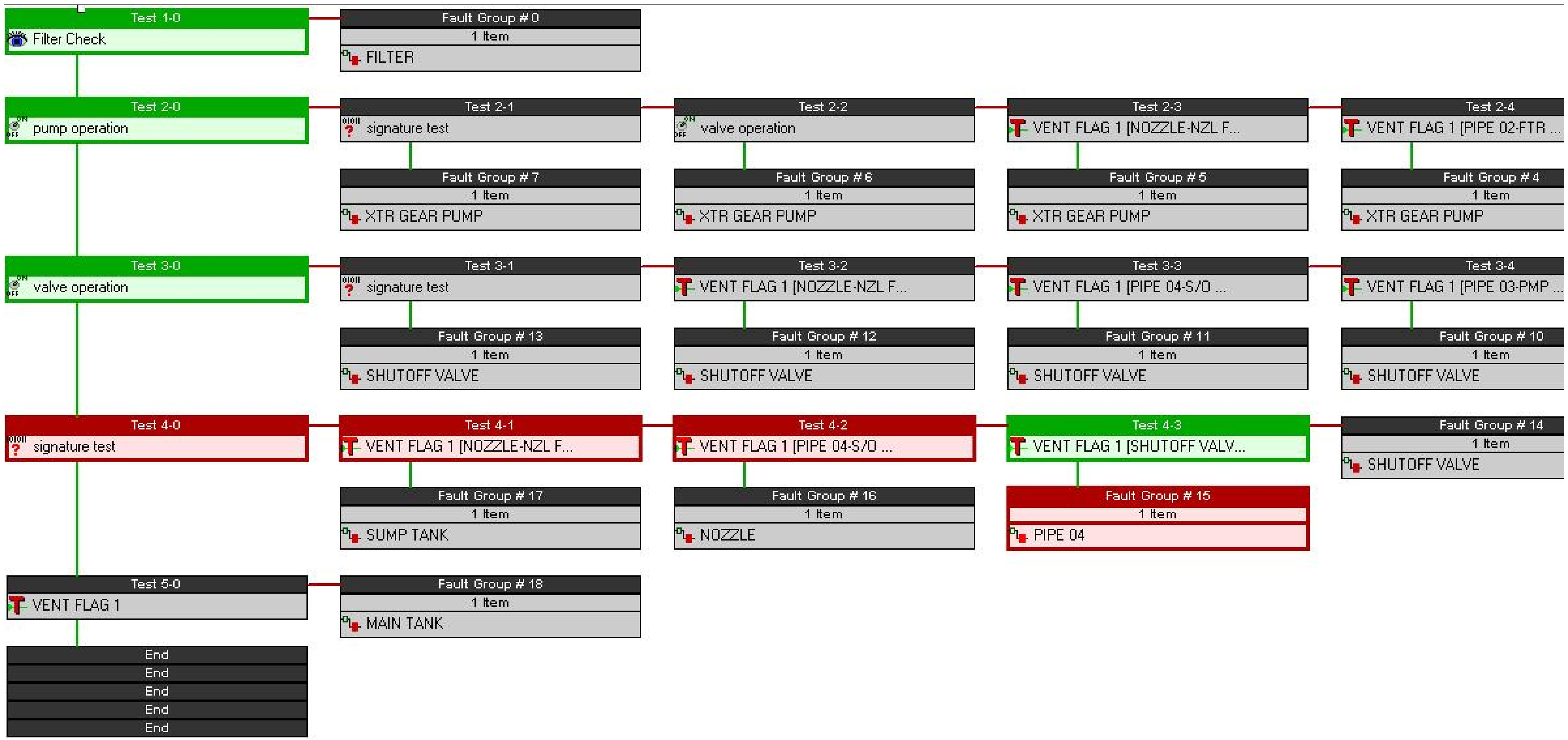


Figure 4-5 DFD with maximise operation and maximise fuction

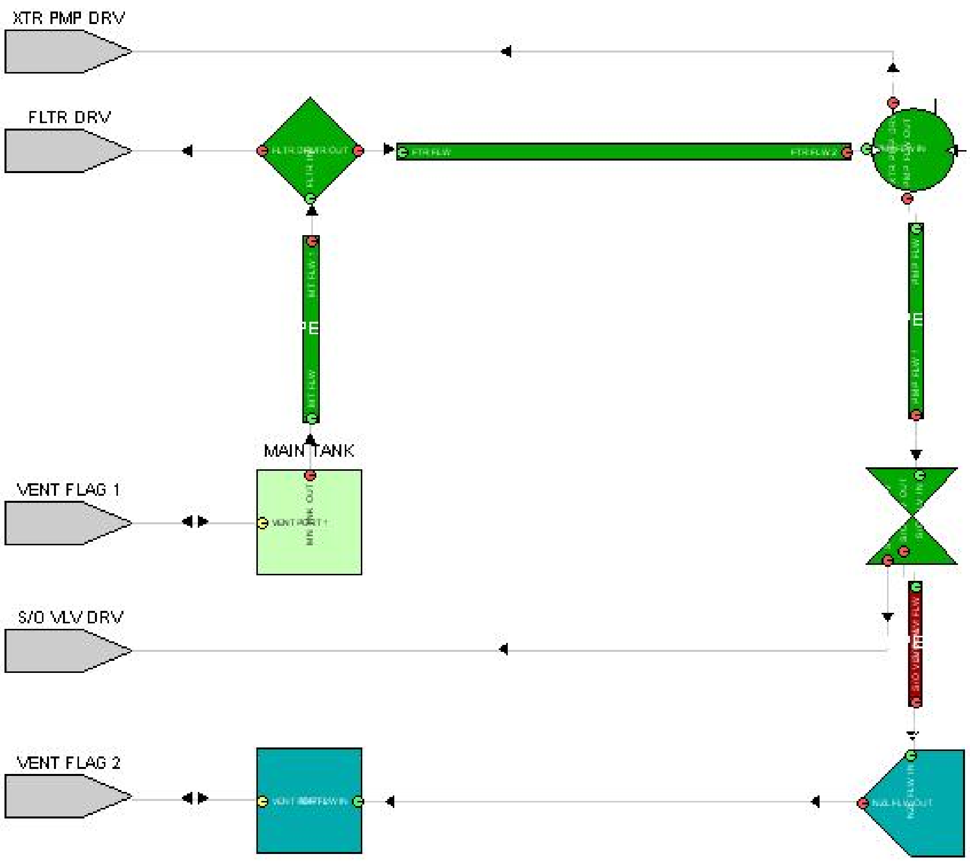


Figure 4-6 System Model with maximum operation and maximised function

4.2 Diagnostic Reports

There are three categories of diagnostic reports available in the pull down reports menu, they are:

- Study report – Study statistics, Diagnostic settings and Aggregate reliability;
- Detection report – Detection order, Detection coverage and item Detection;
- Isolation report – Diagnostic flow table, fault isolation, fault group statistics, test point placement. Subset FD/FI statistics report.

The reports presented below are based on the algorithm shown in Tab. 4-2, based on the mission profile. In this project the detectability with fewer amount of test produces cost efficient prognostics and diagnostics for the fuel system test rig and is helpful to the users.

Detect Malfunction with fewest tests	Multiple-Fault: Half split Failure Probabilities [Refinement Postponed]

Table 4-2 Algorithm used to produce reports

4.2.1 Study Report

4.2.1.1 Study Statistics

This study statistic report provides information about the current diagnostic study. As this report is more similar to that of bill of materials report, in addition to that diagnostic functions are explained and cover the entire study statistics of the model. The sub reports list unused functions and failure modes, design instances, disconnected or improperly linked I/O flags are predominantly useful for testing the accuracy of a hierarchical design. It is shown in the Appendix A.3 Study Statistics.

4.2.1.2 Diagnostic Settings

It provides a listing of the settings associated with the current diagnostic study. It is sub divided into three sections, they are:

- General Settings;
- Detection Settings;
- Isolation Settings.

It mainly comprises of user-selectable parameters used in generating diagnostics (for example: scope, hierarchy settings, test candidates, algorithms, weightings, cut offs). It is explained in Appendix A.4 Diagnostic Settings Report.

4.2.1.3 Aggregate Reliability

In association with the cumulative reliability value of this model, this report provides the individual failure rates, design and assigns according to this model. This is shown in Appendix A.5 Aggregate Reliability Report.

4.2.2 Detection Report

4.2.2.1 Detection Order

It consists of two sections, summary and detection order section, the summary section contains total functions detected (Probability of all functions), total probability detected (Percentage of total failure probability), the aggregate failure design for the system design and the simulated MTBF for the system design.

The detection order section contains, the number of the test in the detection sequence, the name and type of the detection sequence, the name and type of the detection test, the minimum functions detected (percentage of all functions proven individually by that test and cumulatively by all tests up to and including the test), the minimum probability detected (percentage of the total failure probability proven individually by that test and cumulatively by all tests up to and including the test) and the maximum probability detected (percentage of the total failure probability suspected individually by that test and cumulatively by all tests up to and including the test). This is explained in Appendix A.6 Detection Order Report.

4.2.2.2 Detection Coverage

The detection coverage test details the items and output functions detected by that test, replacement cost and time assigned to each detected item, the failure probability for each detection output function, the total replacement cost, replacement time and failure probability for items/functions detected by that test, the stimuli for that test and functions explicitly not detected by that test. It also lists functions not detected/proven by the entire detection sequence and failure modes not detected/proven by the detection sequence. It is shown in Appendix A.7 Detection Coverage Report.

4.2.2.3 Item Detection

In the item-detection report, it lists the very specific detection tests that are used to prove/detect each function in the design. This report is used for trouble shooting diagnostics, when a function is not detected by the expected detection test. It is shown in Appendix A.8 Item Detection Report.

4.2.3 Isolation Reports

The isolation capability of this design is collectively described by five isolation reports. These reports provide valuable information that is used to improve isolation capability. These reports benefit fault group size reduction, optimised placement of test and monitor points, and operational mode coverage.

4.2.3.1 Fault Isolation

This report is primarily comprised of three sections, they are:

- Fault group size statistics: lists the percentage of fault groups of that size that can be isolated using testing; the probability that fault groups of that size that can be isolated using testing; the probability that fault groups of that size can be isolated using both testing and prioritised replacement;

- Cost/time to diagnose a primary failure: provides detailed statistics on the cost and time to isolate, replace and repair a failure. This section provides several useful maintainability measures such as the Mean Time to Repair (MTTR), inherent ability and maintenance ratio.

- Tests to diagnose a primary failure: describes number of tests needed to detect/isolate or diagnose a failure. Appendix A.9 Fault Isolation Report describes it clearly.

4.2.3.2 Fault Group Statistics

The report explains the fault groups based on the following categories:

- Fault Group Size;
- Fault Group by Item;
- Fault group by function;
- Fault group details.

This is explained in Appendix A.10 Fault Group Statistics.

4.2.3.3 Test Point Placement

This report explains ranking of testing locations for the design, ranking of the tests generated for the model, ranking of the test nodes, ranking of the best test locations for the design, locations within the design that do not add detection or isolation capability and generated tests that do not add detection or isolation capability. This report helps the author to optimise test placement and implementation throughout the design and reveal which tests are not providing much benefit. It is shown in Appendix A.11 Test Point Placement Report.

4.2.3.4 Subset FD/FI Statistics

It provides detection, multiple failure and common cause failure statistics for specific operating modes or selected items. In this report a category can be created for each object and the resulting report will show the FD/FI percentages for just those objects. Category creation and definition are useful for models which have multiple levels of hierarchy and require the FD/FI numbers to be calculated for various levels of the design. It is shown in Appendix A.12 FD/FI Statistics by Category Report.

4.2.3.5 Diagnostic Flow Table

A diagnostic sequence of tests created by this report is used for diagnostic documentation and /or troubleshooting. It also lists all fault groups that are isolated within the given diagnostic study. It actually provides the textual representation of the diagnostic flow diagram. It is actually useful in producing malfunction isolation procedures and model verification as it provides the logical and strategic flow of the diagnostics generated for this study. It is explained in Appendix A.13 Diagnostic Flow Table.

4.3 eXpress FMECA

The eXpress FMECA (Failure Mode, Effect and Criticality Analysis): Used to determine the effects that individual failure modes have upon system/subsystem behaviour and the criticality of the resulting effects. Section 4.0 describes how failure mode and effects are added to an eXpress model. These two data elements are necessary to generate a FMECA study in eXpress.

As with diagnostic studies, different FMECA studies can be created to evaluate different reliability scenarios or case studies. To create a FMECA study the user select ‘New’ FMECA study from the file pull-down menu and select the file to be linked and save it in a desired location by giving a name. By selecting the various different data elements to access it in the FMECA charts.

FMECA study provides information about multiple levels of the hierarchical design. The data contained in the report can be used to verify and validate a hierarchical system modelled in eXpress. (eXpress Online Help, 2011). The FMECA chart is attached in the Appendix B.1FMECA Chart.

5 Testability Analysis

Chapter 5 investigates and analyze the results produced by the diagnostic report on the testability software eXpress. However due to limited time, few testability analysis are discussed in this chapter. The diagnostic discussion includes three different approaches of test sets to find the best possible test set for this model. The discussion also includes analyse of cost estimation through the fault isolation report and adding sensor to the model. The last part presents the run time authoring tool java applet pointing out its easy access of the model to the maintenance personnel. The challenges faced during the model creation are encapsulated in a separate section.

5.1 Three different Approaches

Finding a suitable testset for the model with minimum number of test and with maximum fault detection and isolation is more complicated. Benefits of it are the efficiency of the testability increases simultaneously reducing cost and time. Hence a three different test set approach applied on the model as shown in the Tab.5-1 to find the best test set producing maximum fault detection and isolation.

Model	Test Set 1	Test Set 2	Test Set 3
First Position	Filter check	UI Filter check	Filter check
Second Position	Pump operation	Pump operation	Pump operation
Third position	Valve operation	ValveInspection	Valve operation
Fourth Position	Fuel flow	Fuel flow	Fuel Probe Test
Fifth Position	----	----	Fuel flow

Table 5-1Three Different Test Sets

Tests	Detection Statistics				Isolation Statistics		
	CUM		CUM FP		Probability of detection	Probability of Isolation	Isolation Effective-ness
	TFD %	TFD(I) %	TFD %	TFD(I) %	%	%	%
Filter Check	14.21	14.21	14.21	14.21	14.21	100	100
Filter Check Pump Operation	42.86	42.86	42.86	42.86	42.86	100	100
Filter Check Pump Operation Valve operation	64.29	64.29	64.29	64.29	64.29	100	100
Filter check Pump operation Valve operation Fuel flow	100	100	100	100	100	100	100

Table 5-2Test set Approach 1

Tests	Detection Statistics				Isolation Statistics		
	CUM		CUM FP		Probability of detection	Probability of Isolation	Isolation Effective-ness
	TFD %	TFD(I) %	TFD %	TFD(I) %	%	%	%
UI Filter Check	7.69	7.69	7.69	7.69	7.69	100	100
UI Filter Check Pump Operation	15.38	15.38	15.38	15.38	15.38	100	100
UI Filter Check Pump Operation ValveInspection	30.77	30.77	30.77	30.77	30.77	100	100
UI Filter check Pump operation ValveInspection Fuel flow	100	100	100	100	100	100	100

Table 5-3 Test set Approach 2

Tests	Detection Statistics				Isolation Statistics		
	CUM		CUM FP		Probability of detection	Probability of Isolation	Isolation Effective ness
	TFD %	TFD(I) %	TFD %	TFD(I) %	%	%	%
Filter Check	14.21	14.21	14.21	14.21	14.21	100	100
Filter Check Pump Operation	42.86	42.86	42.86	42.86	42.86	100	100
Filter Check Pump Operation Valve operation	64.29	64.29	64.29	64.29	64.29	100	100
Filter Check Pump Operation Valve operation Fuel Probe Test	92.31	92.31	92.31	92.31	92.31	100	100
Filter check Pump operation Valve operation Fuel Probe Test Fuel flow	100	100	100	100	100	100	100

Table 5-4 Test set Approach 3

Tab. 5-2, 5-3, 5-4 shown above represents the reports of detection statistics and isolation statistics of the testability model for three different test sets. In the detection statistics it is divided into Cumulative Percentage (CUM) and Cumulative Fault Percentage (CUM FP). This percentage will further subdivided into Total fault detection (TFD) and total fault detection with interference (TFD (I)).

The numerical values are based upon the test coverage, failure modes, states and failure effects given in the model. For the tests in the first position the total fault detection with and without interference and probability of detection values are shown in the Fig. 5-1to 5-4, which is very low and indicated its coverage percentage to the overall model. Tests are added till the maximum detection and isolation percentage of the model are found.

First step in fault detection and isolation is to capture faults in the filter, as filter is the first component which is positioned after the main tank, will be subjected to debris and contaminants. Inspection test is used to visually inspect the health status of the filter and the test location to be allocated for this is filter output flag. The health condition of the filter is obtained through a sensor. Hence filter inspection test is sufficient to find the fault detection and isolation for this model. The overall detectability of the model after inserting the filter inspection test gives a value of 14.21%. This value depends upon the amount of coverage that the inspection tests do. The third test set contains the same inspection test for filter hence the value also be the same.

In a different approach, if user initiated test is used instead of filter inspection test, user initiated test is inserted in the first position of the second test set model. The overall detection and isolation coverage values of the software produced due to this test is 7.69%. The variation is mainly due to the coverage and the selected object states of each different test. However, inspection test is finalised for the filter component as the coverage is more accurate by concentrating only the filter and its object states and has no external stimuli.

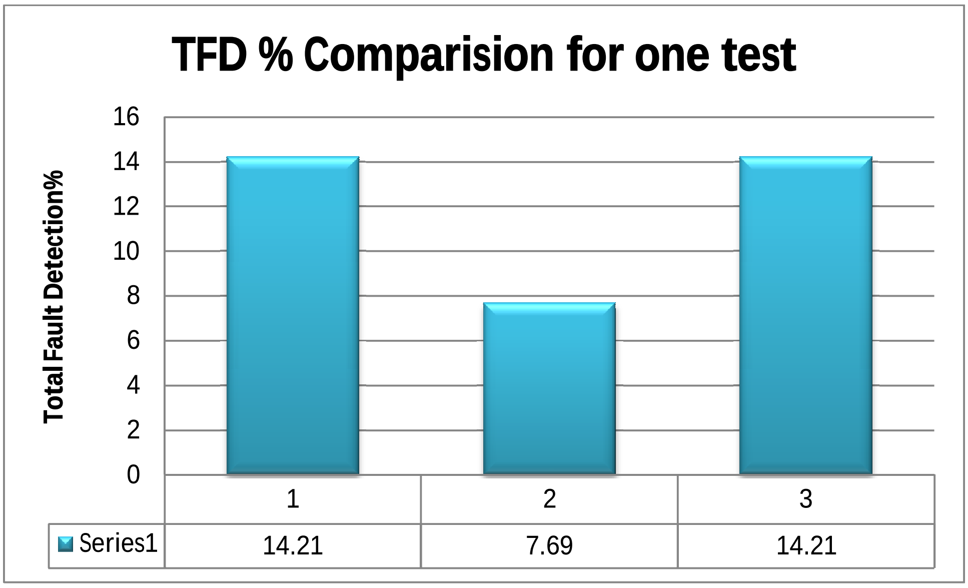


Figure 5-1 TFD% of three different test sets for test in the first Position

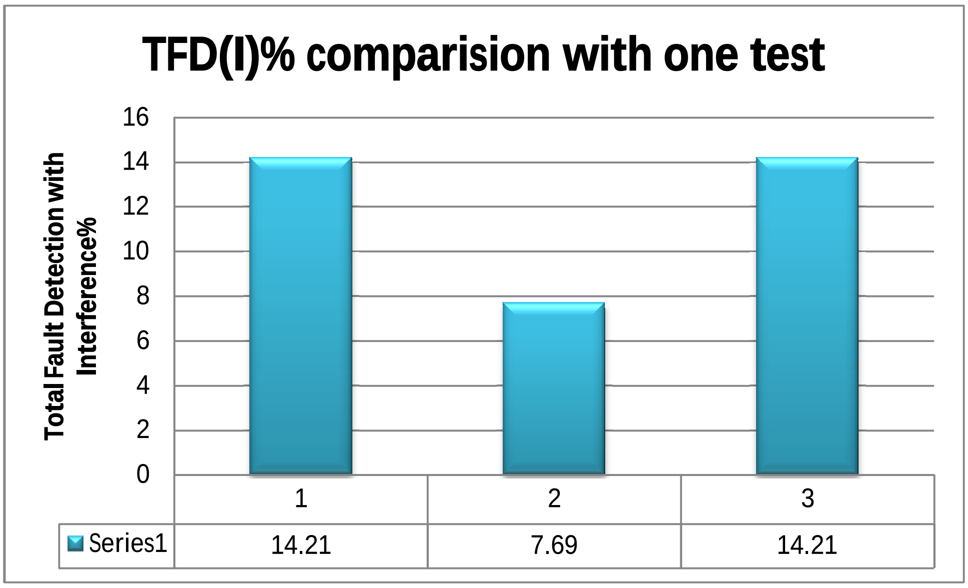


Figure 5-2 TFD (I) % of three different test sets for test in the first Position

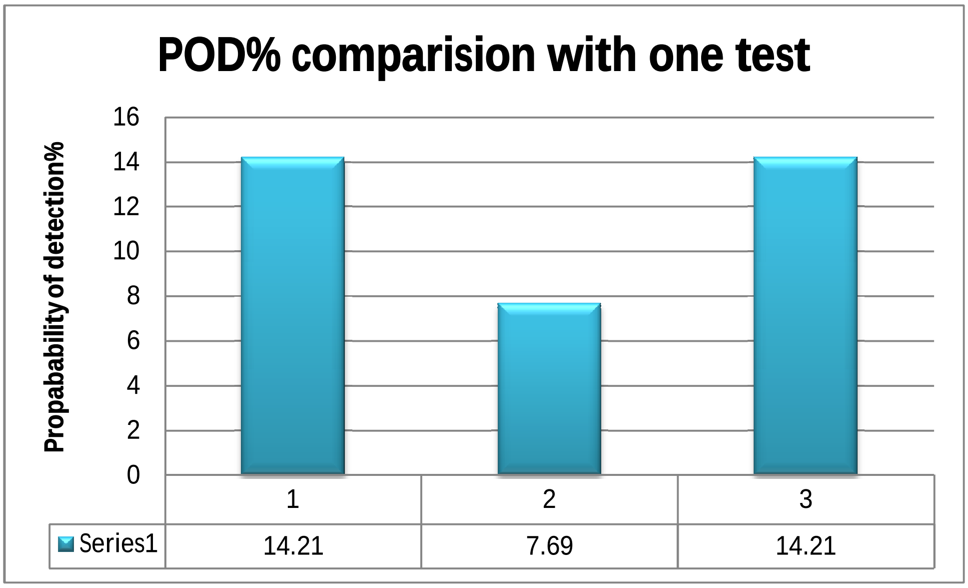


Figure 5-3 POD% of three different test sets for test in the first Position

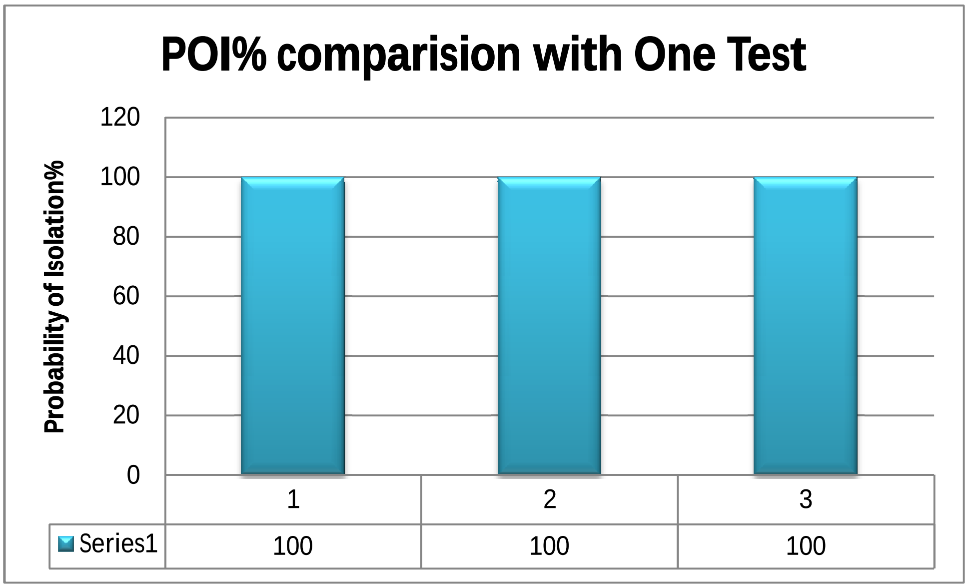


Figure 5-4 POI% of three different test sets for test in the first Position

Fig. 5-1 to 5-4 indicate the variation of the values of the test in first position of three test set model. In the test set 1 and 3 filter inspection test is taken into account and in the second test set model user initated filter inspection test is used. Hence the variation arises. TFD indicates total fault detection and TFD(I) indicates total fault detection with interference, where the interference refers to the failure mode and the failure effect.

Next component which is positioned after the filter is the pump, which operates at a constant rpm. The failure of the component mainly affects its operation; hence operational test is inserted into the component pump. The operational test as explained in the previous chapter **Error! Reference source not found.** concentrates mainly on the failure mode and its functions. Operational test captures the operational sequence of the pump to be good or bad. Hence by applying operational test, the coverage and the total fault detection percentage it covers will be 28.65% which comes from the difference between the overall cumulative percentage 42.86% with the previous cumulative percentage 14.21% produced by the filter inspection test. The term cumulative means increasing in quantity, likewise by adding tests the fault detection percentage also increases. However, user initiated test also inserted to check for the detection coverage for the reason to test the model and know the values similar to that of operational test and the value obtained will be 15.39% and cumulative percentage will be 30.77%. As the pump mainly concentrate on operational performance operational test is the best sequence to be used to find fault detection and isolation. Hence, Operation test must be finalized for the final model of the fuel system test rig.

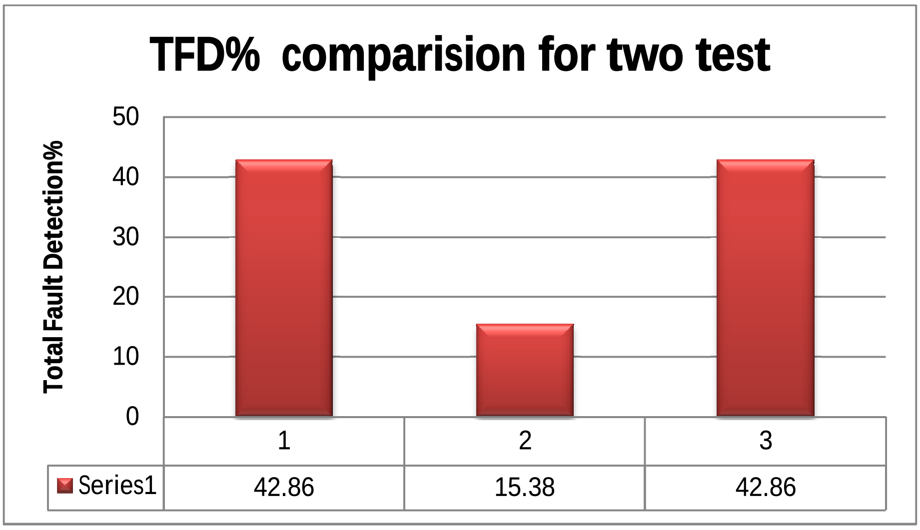


Figure 5-5 TFD% of three different test sets for test in the second Position

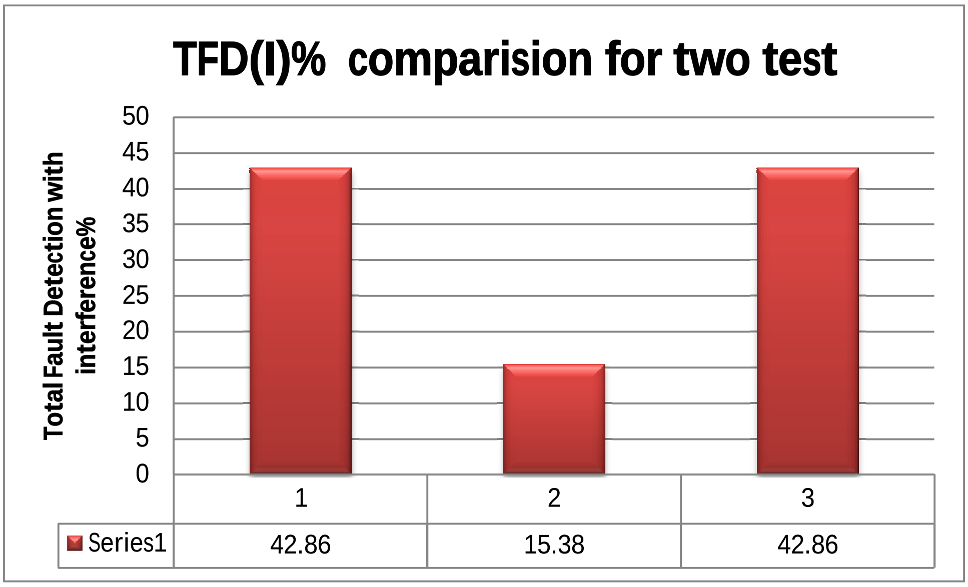


Figure 5-6 TFD (I) % of three different test sets for test in the second Position

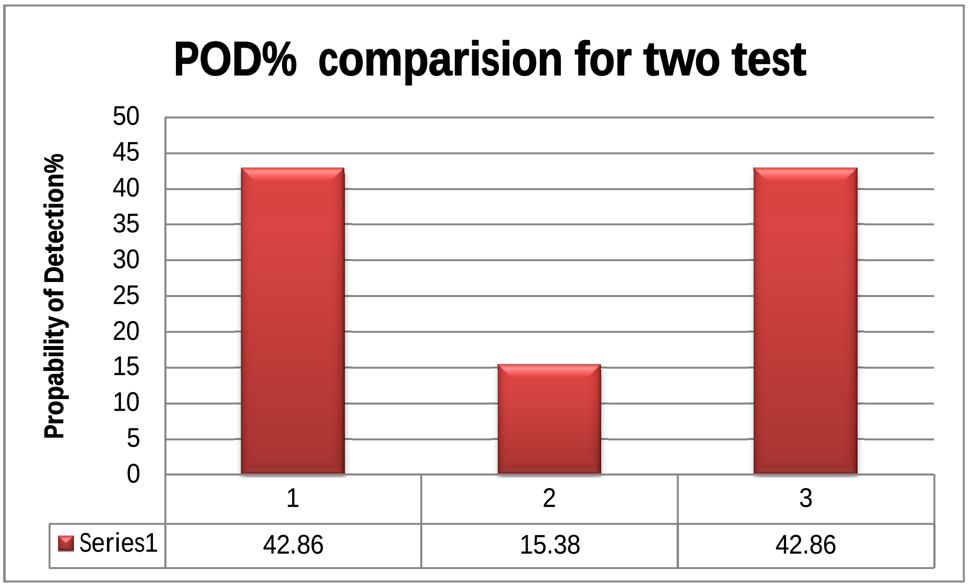


Figure 5-7 POD% of three different test sets for test in the second Position

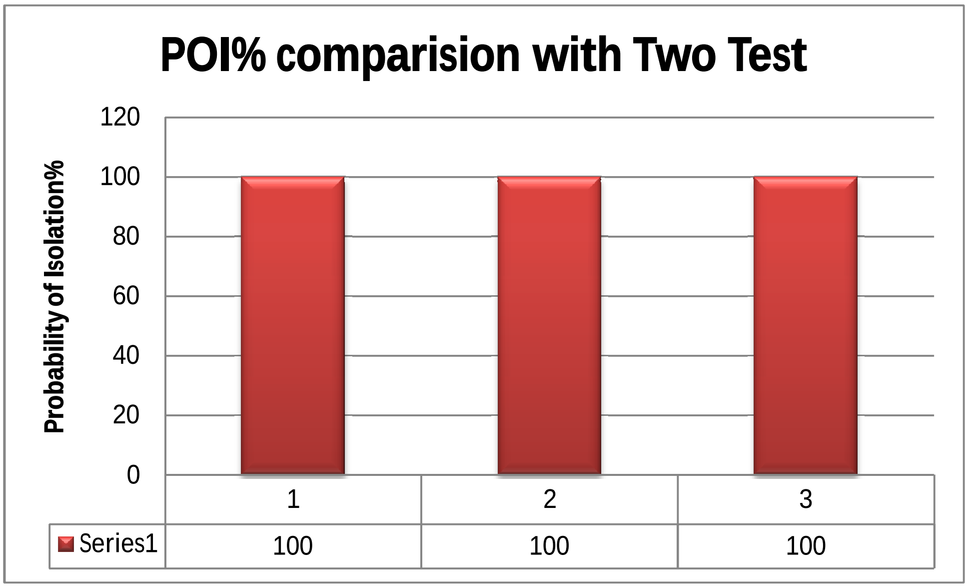


Figure 5-8 POI% of three different test sets for test in the second Position

Fig. 5-5 to 5-8 indicate the variation of the values of the test in second position of three test set model. In the test set 1 and 3 pump operational test is taken into account and in the second test set model user initated operational test is used which gives rise to the variation.

The next component which is positioned after the pump must be shutoff valve which open and closes as its operation. Thus an operational test is applied to the shutoff valve drive which covers the functional mode and its effects particularly on the valve. Hence by applying operational test, the coverage and the total fault detection percentage it covers will be 21.43% which comes from the difference between the overall cumulative percentage 64.29% with the previous cumulative percentage 42.86% produced by the pump operation test. In another case there is possibility of inspecting the shut off valve, hence there is a possibility of applying inspection test. By applying inspection test the detection and isolation statistics value will be 15.39% and the cumulative percentage will be 30.77%. However in this model, onboard diagnostics is more important than ground maintenance, hence operational test is finalised for the final model.

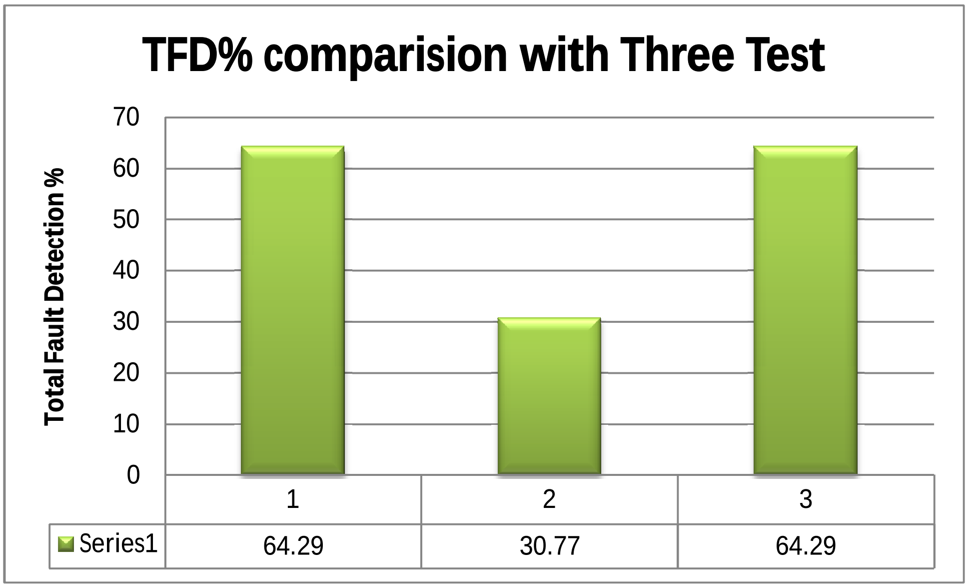


Figure 5-9 TFD% of three different test sets for test in the third Position

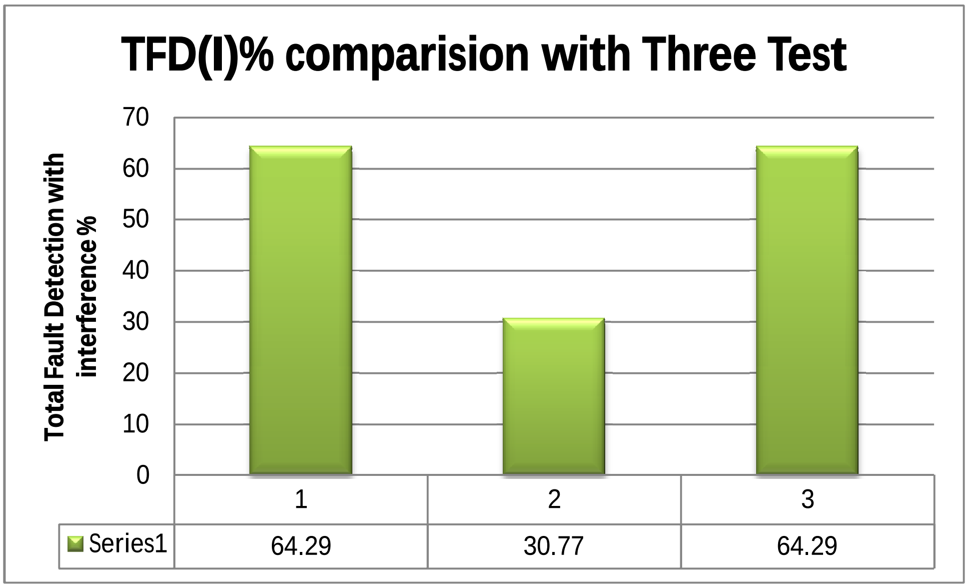


Figure 5-10 TFD (I) % of three different test sets for test in the third Position

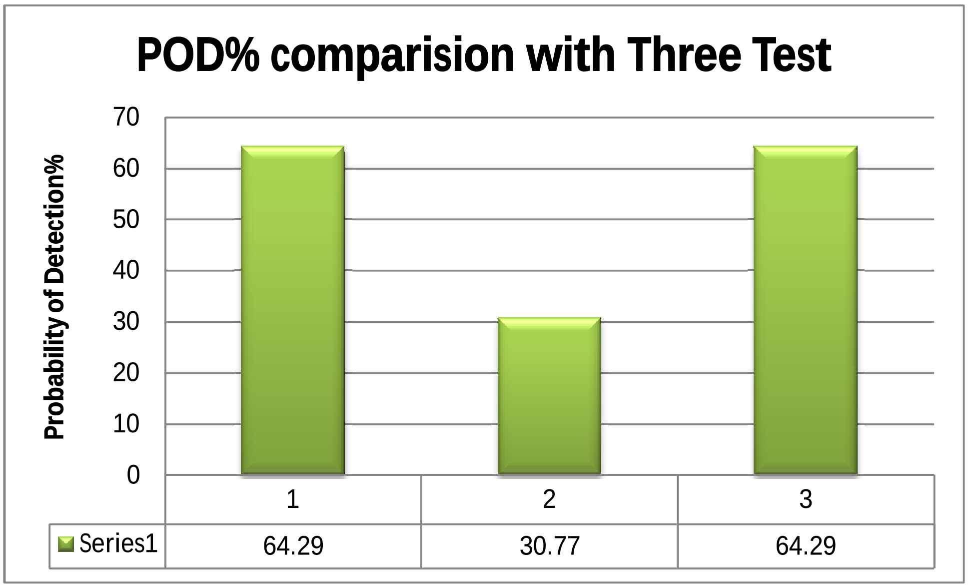


Figure 5-11 POD% of three different test sets for test in the third Position

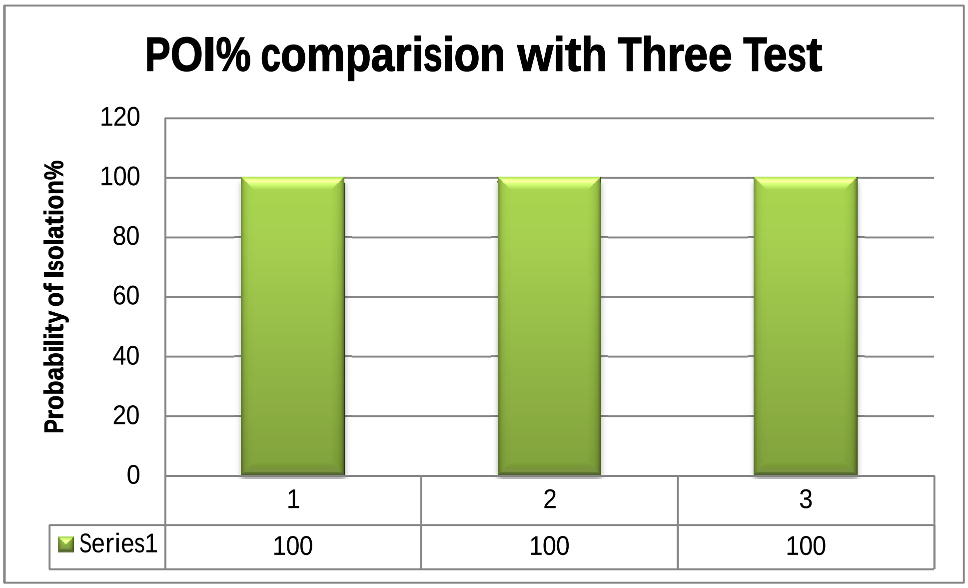


Figure 5-12 POI% of three different test sets for test in the third Position

Fig. 5-9 to 5-12 indicate the variation of the values of the test in third position of three test set model. In the test set 1 and 3, S/O Valve operational test is taken into account and in the second test set model S/O valve inspection test is used. This has given rise to the variation.

Once the three tests are applied it is seen that the fault detection and isolation percentage of the major components are finished. Now signature test is applied to the overall model in which the test stimuli must be the last output flag, so that the entire object/component will be subjected into this test. While creating signature test itself the author brought all the objects under the test, hence the coverage is 100% and every function and failure mode will be included in the signature test.

Once this test is inserted the total fault detection, percentage will increases to 100 % (maximum). Since this signature test will cover all the functions, there will be a question about whether this test is enough for the entire testability model and the requirement of other tests for major individual component. The answer for this is explained in the previous chapter **Error! Reference source not found.**. However this signature test covers the entire failure mode, design and object states its efficiency is not good while using it as the primary and only test for the testability model.

In addition to this, probe test is an option to include in this model which will add the final output of the model results. Using probe test, confirmation of the results of the other test is possible and this will lead to higher accuracy. Hence by applying probe test, the coverage and the total fault detection percentage it covers will be 28.02% which comes from the difference between the overall Cumulative percentage 92.31% with the previous cumulative percentage 64.29% produced by the filter inspection test.

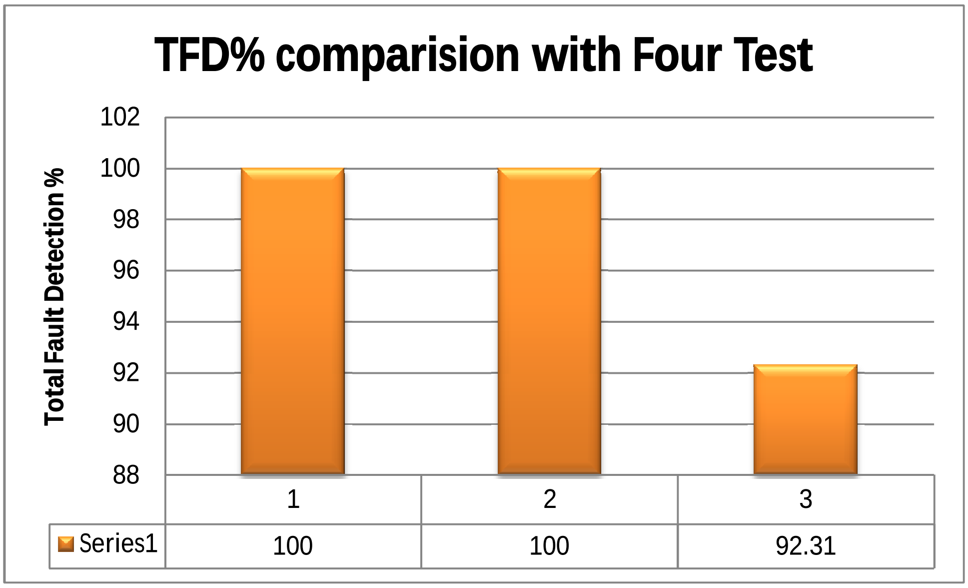


Figure 5-13 TFD% of three different test sets for test in the fourth Position

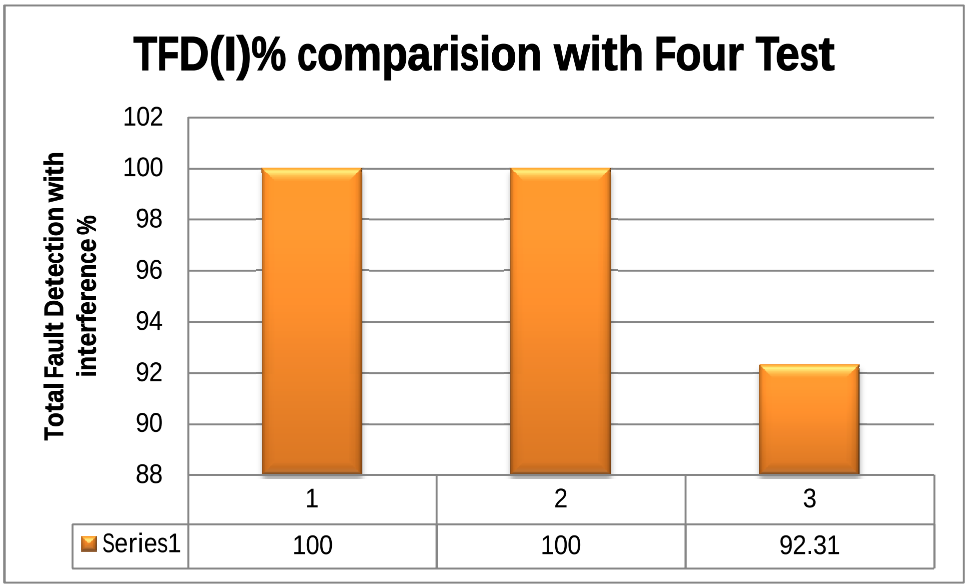


Figure 5-14 TFD (I) % of three different test sets for test in the fourth Position

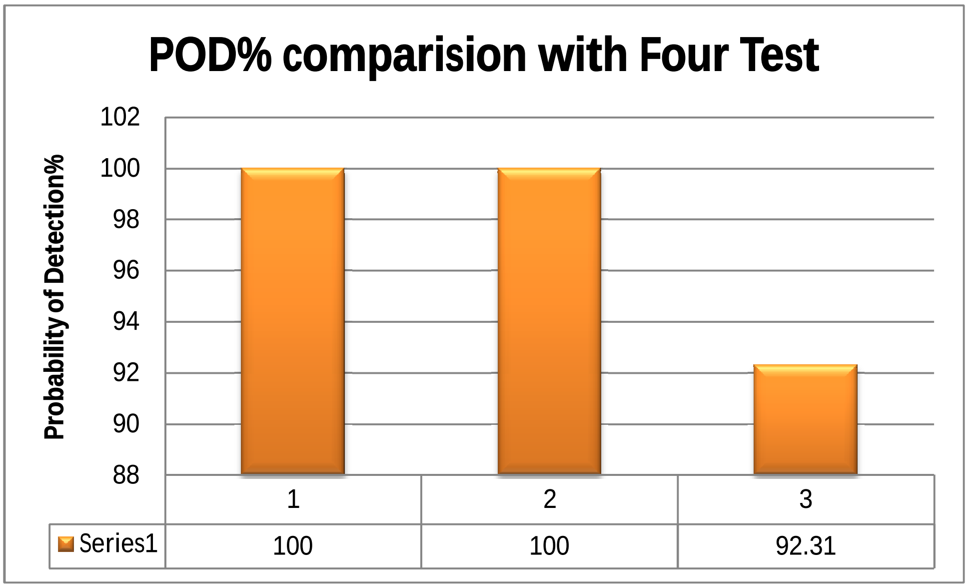


Figure 5-15 POD% of three different test sets for test in the fourth Position

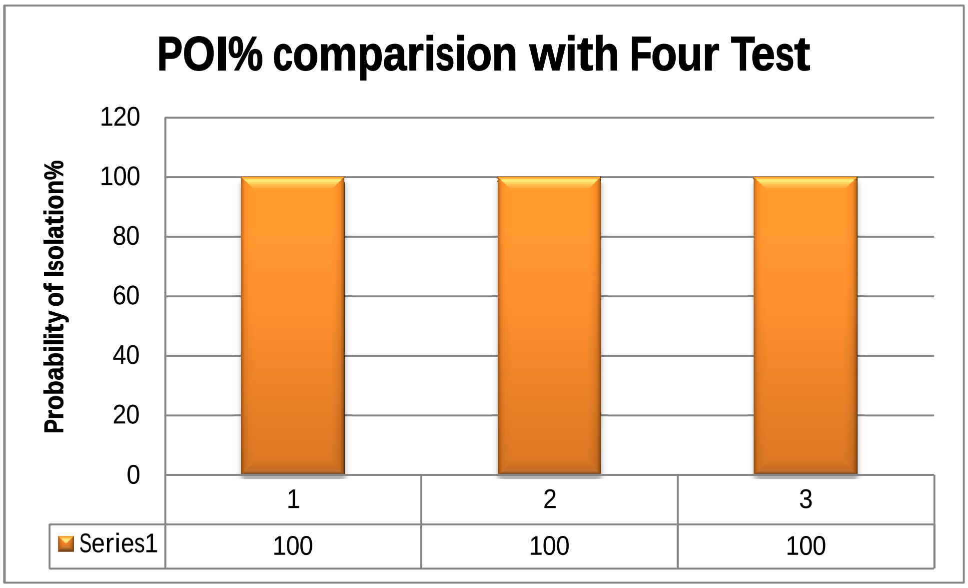


Figure 5-16 POI% of three different test sets for test in the fourth Position

Fig. 5-13 to 5-16 signify the variation of the values of the test in fourth position of three test set model. In the test set 1 and 2 fuel flow signature test is taken into account and in the second test set model fuel flow probe test is used. Hence the variation arises. In the Percentage of Isolation graph only the values shown are maximum and similar in all the tests because in each test the corresponding object or component fault is isolated separately. So it always has the 100 percent value.

Through the three different approaches shown in Tab 4-1, test set model 1 is taken to the final model which has the algorithm of detection and isolation of system faults with fewer tests. All the reports are calculated based on these four tests only.

5.2 Cost and Time factor Prediction

In the eXpress software, apart from this model designing and testability process, attribute calculation plays an important role. The cost, reliability and time prediction reports helps the designers and maintenance people to know factors influencing fault isolation like cost to isolate, replace and repair and time to isolate, replace and repair components.

In attribute section of the component details panel, the user defines the cost, time and reliability value. The reliability value can be entered in two ways either by failure mode or through Mean Time between Failure (MTBF). This report offers details to the maintenance personnel about the overall limit of cost, time and reliability of the entire fuel system by calculating it from the individual components. A higher positive differential value of time and cost to isolate, replace and repair shows that the cost distribution among the model is good and the component and model gives the maximum benefit to the customer and if it shows a negative value it indicates that the cost and time utilised must be additional and cause loss to the customer.

The user defined reliability, cost and time values of the components to get the maximum benefit to the maintenance personnel through the model is given in the Tab. 5-5:

Component	*Cost* ($)	Time (Minutes)	Reliability (Years)
Main tank	100	100	10
Filter	90	90	5
XTR Gear Pump	90	90	5
Shut off valve	90	90	10
Nozzle	90	90	10
Sump tank	25	30	10
Pipe	10 each segment	30 each segment	10 each segment

Table 5-5 Cost, Time and Reliability value for model 1

Cost to Isolate (in US Dollars)			**Time to Isolate** (in minutes)		
Minimum:	**2.00**		Minimum:	**1800.00**	
Maximum:	**2.00**		Maximum:	**1800.00**	
Average:	**2.00**		Average:	**1800.00**	
Expected (MCTI):	**2.00**		Expected (MTTI):	**1800.00**	
Differential:	**0.00**	**(0.00%)**	Differential:	**0.00**	**(0.00%)**
per Operating Hour:	**0.0003**		per Operating Hour:	**0.2464**	
Cost to Replace (in US Dollars)			**Time to Replace** (in minutes)		
Minimum:	**20.00**		Minimum:	**1800.00**	
Maximum:	**100.00**		Maximum:	**6000.00**	
Average:	**60.45**		Average:	**3872.73**	
Expected:	**62.08**		Expected:	**3950.00**	
Differential:	**1.63**	**(+2.69%)**	Differential:	**77.27**	**(+2.00%)**
per Operating Hour:	**0.0085**		per Operating Hour:	**0.5407**	
Cost to Repair (in US Dollars)			**Time to Repair** (in minutes)		
Minimum:	**22.00**		Minimum:	**3600.00**	
Maximum:	**102.00**		Maximum:	**7800.00**	
Average:	**62.45**		Average:	**5672.73**	
Expected (MCTR):	**64.08**		Expected (MTTR):	**5750.00**	
Differential:	**1.63**	**(+2.61%)**	Differential:	**77.27**	**(+1.36%)**
per Operating Hour:	**0.0088**		per Operating Hour:	**0.7871**	

Table 5-6 Multiple Fault Isolation (Model 1)

The Fault Isolation report generated based on the above values is given in the Tab. 5-6. The description of each term is given in the section 5.2.1. The author selected an another set of cost, time and reliability value to compare the difference between the model 1 and 2 by changing the reliability value of each component they are given in the Tab. 5-7 below

Component	*Cost* ($)	Time (Minutes)	Reliability (Years)
Main tank	200	150	20
Filter	50	60	10
XTR Gear Pump	60	50	5
Shut off valve	40	60	7
Nozzle	90	50	5
Sump tank	50	60	20
Pipe	10 each segment	20 each segment	10 each segment

Table 5-7 Cost, Time and Reliability value for model 2

The fault isolation report generated based on the above values is shown in the Tab. 5-8.The discussion and analysis of the readings amongst the two models is described in the section 5.2.2 Comparison of the Fault Isolation Reports.

Cost to Isolate (in US Dollars)			Time to Isolate (in minutes)		
Minimum:	**2.00**		Minimum:	**1800.00**	
Maximum:	**2.00**		Maximum:	**1800.00**	
Average:	**2.00**		Average:	**1800.00**	
Expected (MCTI):	**2.00**		Expected (MTTI):	**1800.00**	
Differential:	**0.00**	**(+0.00%)**	Differential:	**0.00**	**(+0.00%)**
per Operating Hour:	**0.0003**		per Operating Hour:	**0.2347**	
Cost to Replace (in US Dollars)			**Time to Replace** (in minutes)		
Minimum:	**10.00**		Minimum:	**1200.00**	
Maximum:	**200.00**		Maximum:	**9000.00**	
Average:	**66.36**		Average:	**3763.64**	
Expected:	**50.06**		Expected:	**2943.75**	
Differential:	**-16.30**	**(-24.56%)**	Differential:	**-819.89**	**(-21.78%)**
per Operating Hour:	**0.0065**		per Operating Hour:	**0.3838**	
Cost to Repair (in US Dollars)			**Time to Repair** (in minutes)		
Minimum:	**12.00**		Minimum:	**3000.00**	
Maximum:	**202.00**		Maximum:	**10800.00**	
Average:	**68.36**		Average:	**5563.64**	
Expected (MCTR):	**52.06**		Expected (MTTR):	**4743.75**	
Differential:	**-16.30**	**(-23.84%)**	Differential:	**-819.89**	**(-14.74%)**
per Operating Hour:	**0.0068**		per Operating Hour:	**0.6185**	

Table 5-8 Multiple Fault Isolation (Model 2)

The user defined cost and time values for test are assumed as per the current industrial prices of the similar models. The Cost and time values for the test to be used in the testability model is given in the Tab. 5-9 below:

Test	Cost $	Time (Minutes)
Operational Test	1	30
User-Initiated Test	1	30
Signature Test	2	30
Inspection Test	1	30
Probe Test	2	30

Table 5-9 Cost and Time value for the test

The user defined cost and time value for the tests shown in the Tab. 5-9 does not have a large impact on the cost/time factor of the fault isolation. Hence the reliability value for cost and time for the test given is the same for two different models. In the fault isolation report shown in the Appendix A.9 Fault Isolation Report, the multiple failure fault group size statistics section contains metrics that describe the fault isolation capability of the diagnostics and design in the current diagnostic study.

In the report, the fault group size percentages section of the fault isolation report, lists the different fault group size. The fault group size is isolated by the diagnostic sequence in the current diagnostic study. In this model the fault group size is calculated as 11, which is shown in the Fig. 5-17 below

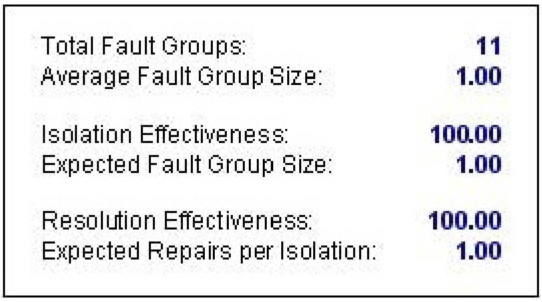


Figure 5-17 Fault Group Count

The Isolation effectiveness describes how good the current diagnostic study diagnostic sequence is able to isolate a fault group containing a single repair item. For this model it is calculated as 100 % and it shows that this model is 100% capable to isolate a fault group containing a single repair item.

The expected and average fault group size indicates the expected and average number of replaceable items in fault groups isolated in the current diagnostic study. For this model it was calculated to give a value of 1.

5.2.1 Description of Fault Isolation report

In the Fault Isolation Report shown in Tab. 5-6 and 5-8, cost/time to diagnose a primary failure using testing only section contains metrics that describe cost and time burden associated with isolating a failure and/or replacing the items in the isolated fault group.

The metrics that appears in this section are versions of standard maintainability metrics that have been simplified so that it can provide effective feedback in the earlier design phase. So the analyst can estimate the impact it changes to a diagnostic design upon system maintainability.

Minimum

Minimum Time/Cost to isolate is the minimum cumulative test time/cost with detecting and isolating a failure in the fault group in the diagnostic sequence.

Minimum Time/Cost to Repair is the minimum time/cost linked with isolation and replacement of all items in the fault group isolated by the diagnostic sequence.

Minimum Time/Cost to Replace is the minimum aggregate object replacement time/cost linked with replacement of all the items in the fault groups isolate by the current diagnostic sequence.

Maximum

Maximum Time/Cost to Isolate is the maximum cumulative test time/cost with detecting and isolating a failure in the fault group in the diagnostic sequence.

Maximum Time/Cost to Repair is the maximum time/cost linked with isolation and replacement of all items in the fault group isolated by the diagnostic sequence.

Maximum Time/Cost to Replace is the maximum aggregate object replacement time/cost linked with replacement of all the items in the fault groups isolate by the current diagnostic sequence.

Average

Average Time/Cost to isolate is the average time calculated by the summation of time/cost linked with isolation and replacement of each isolated fault group and divides it by total number of isolated fault group

Average Time/Cost to Repair is the average time calculated by the summation of time/cost linked with isolation and replacement of each isolated fault group and divided by total number of isolated fault group.

Average Time/Cost to Replace the average time calculated by the summation of aggregate replacement time/cost linked with each isolated fault group and divided by total number of isolated fault group.

Expected

Expected Mean Time to Isolate/ Mean Cost to Isolate (MTTI/MCTI) is calculated by multiplying each Fault groups time/cost to isolate by its aggregate failure probability and add the results for all the isolated ones.

Expected Mean Time to Repair/ Mean Cost to Repair (MTTR/MCTR) is calculated by multiplying each Fault groups combined isolation and replacement time/cost to isolate by its aggregate failure probability and add the results for all the isolated ones.

Expected Mean Time to Replace/ Mean Cost to Replace (MTTR/MCTR) is calculated by multiplying each Fault groups aggregate replacement time/cost to isolate by its aggregate failure probability and add the results for all the isolated ones.

Differential

The Time/Cost to Isolate/Repair/Replace differential is calculated by subtracting average with the expected. (A high positive value indicate that they are well segregated and negative value indicate that they are not well segregated)

Per operating hour

The Time/Cost to Isolate/Repair/Replace per Operating Hour is calculated by dividing the Expected Time/Cost to Isolate/Repair/Replace by the calculated system Mean Time between Failure (MTBF).

Mean Time between Failure (MTBF)

Mean time between failures is the time predicted between Inherent failures of a system during operation.

5.2.2 Comparison of the Fault Isolation Reports

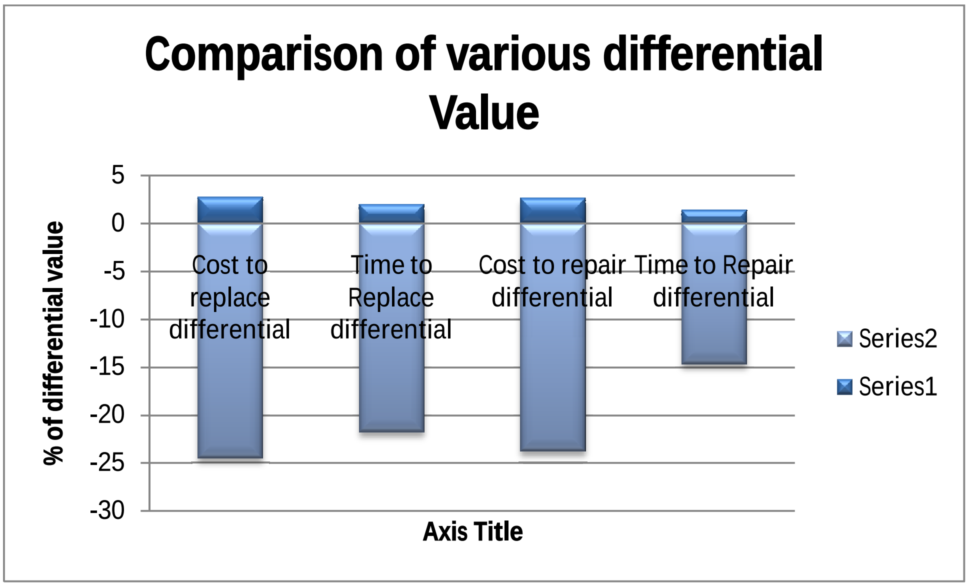


Figure 5-18 Multiple Failure Isolation Differential value

Based on the two different attribute values, the above Fig. 5-18 depicts the variation of positive and negative differential values of cost/time to replace and repair. As series 1 explains the model approach shown in the Tab. 5-4 and series 2 explains the model approach shown in the Tab. 5-5. From the above explanation, it is obvious that the model 1 has provided a positive differential value and indicates the maximum cost and time benefit and the model 2 shows a high negative value which indicate the cost and time factor will exceed the models capability and this produce a loss to the user if the model is implied.

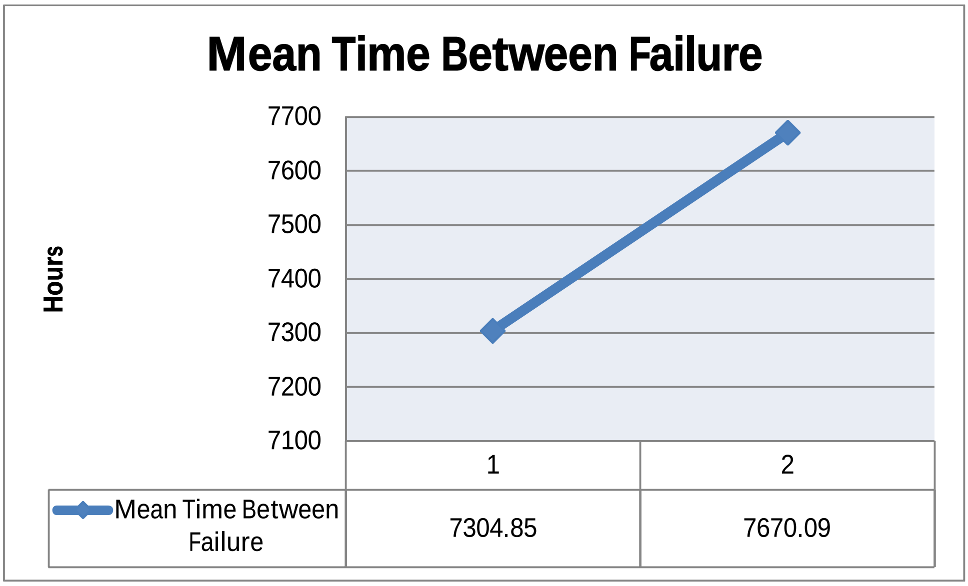


Figure 5-19 Mean Time Between Failure (MTBF) between two models

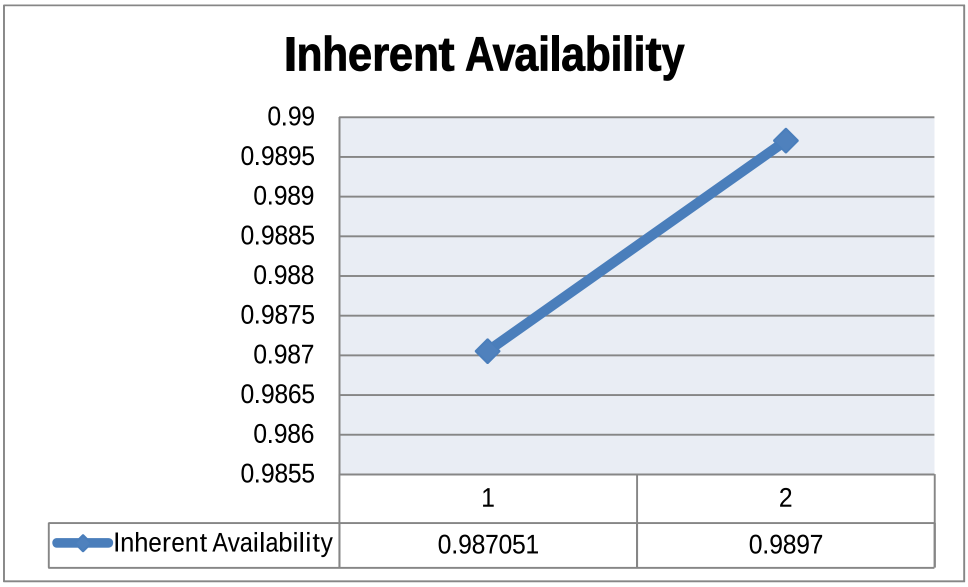


Figure 5-20 Inherent Availability between two models

Fig. 5-19 indicates variation of the two models 1 and 2. The value is calculated based upon the MTBF values given to each component. For the model one, the value obtained is 7304.85 hours and for the model two the value obtained is 7670 hours. It is seen that the MTBF value of model 2 is lesser than that of the model 1. But the difference is not too high and based upon the cost/time to isolate, replace and repair, Model 1 is best suited for this system.

Fig. 5-20 indicates the inherent availability of the system. For the models 1 and 2 the inherent availability difference is not too high and the values represent very close to 1 show the availability of the system is good, based upon the values given in the model 1 and 2.

5.3 Java Applet for the Users

The eXpress java applet is a eXpress run time authorising tool which displays the files in any web browser for the easy access to the user especially maintenance personnel. It is a fully graphical, fully hierarchical and fully sharable file format with three viewing modes. The three viewing modes are design view, design details panel and test coverage view (eXpress Java Applet, 2011).

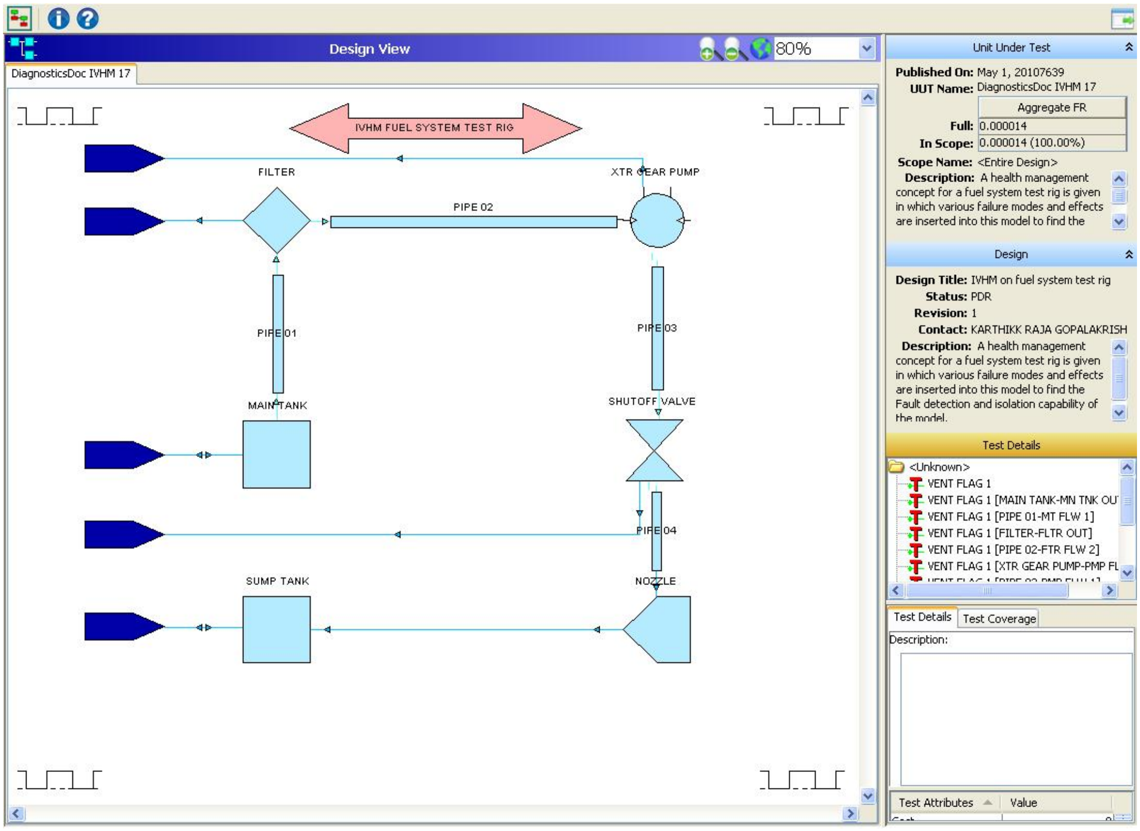


Figure 5-21 Design View of eXpress java applet

Fig. 5-21 describes the design view of the eXpress java applet. In this design view it displays eXpress design similar to that appears on the software. The user can access the lower level design levels like test details, failure mode functional details and object details in the design details panel.

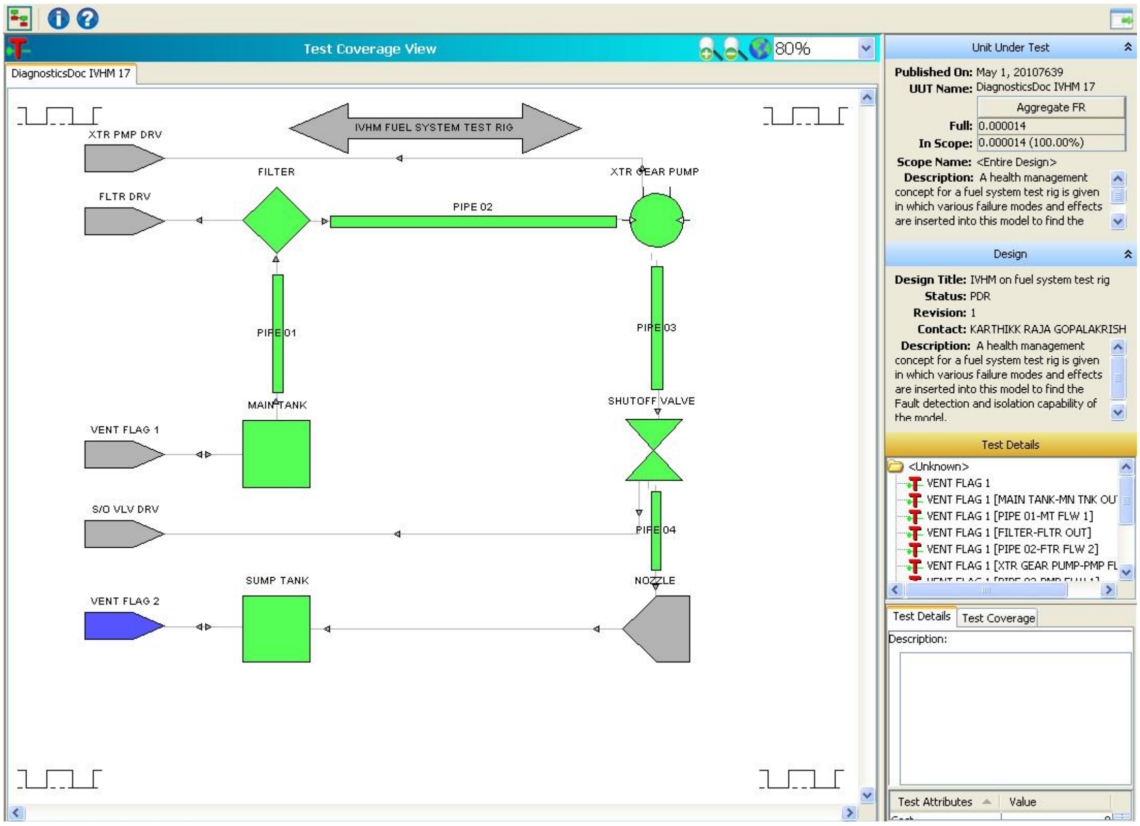


Figure 5-22 Test Coverage View

Fig. 5-22 explains the test coverage view of the eXpress java applet. The colour coded objects and I/O flags displayed in the design sheet window are colour coded to show the coverage of the test selected by the user either in the design details panel or in the diagnostic test sequence. The specific functions or failure modes of a specific test are shown in the design details panel.

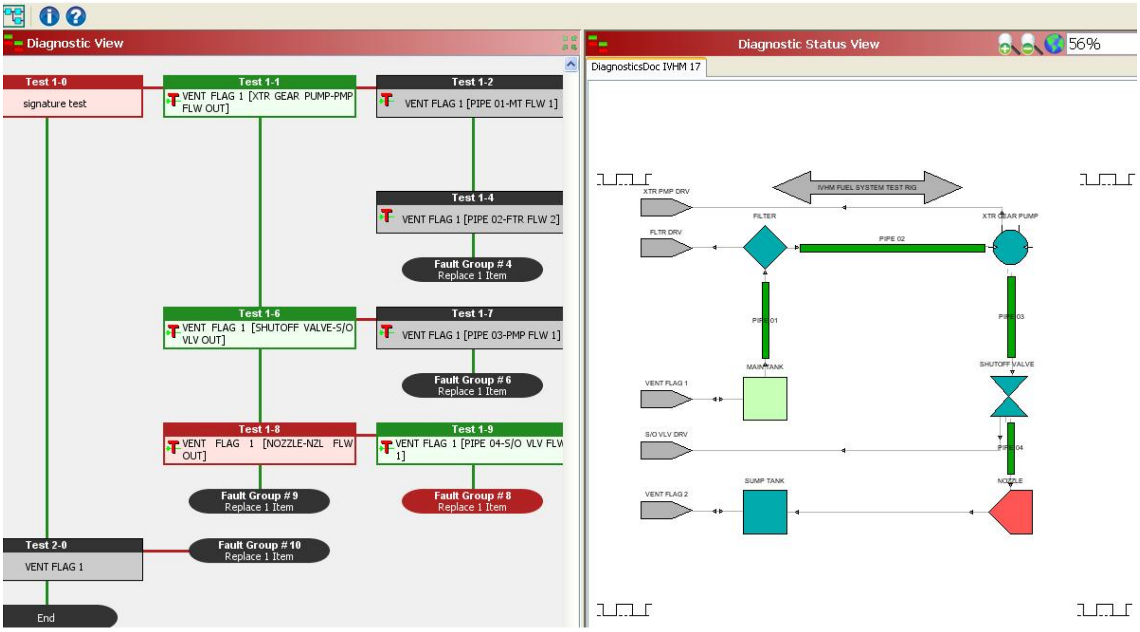


Figure 5-23 Diagnostic Status View

The diagnostic test sequence displayed as a tree on the left hand side of the browser as shown in the Fig. 5-23 depicts the diagnostic status view. In this view the user can select any test or fault group node to see the diagnostic status associated with that node.

In the diagnostic details panel the information about either a selected node or about the designs overall diagnostic capability can be seen.

6 Conclusion and Future Work

The prognosis and diagnosis in civil and military aviation is now grown beyond line maintenance and ground maintenance. Now it is being applied to a full scale onboard maintenance to mitigate the failures happening to the modern unmanned aerial vehicle during flight conditions. Various technologies have arrived to find prognosis and diagnosis of a fuel system test rig but this IVHM technology has a very unique capability.

The reports generated in this project helps the fuel system model to detect and isolate the failures effectively and make use of it to implement on a real test rig. The cost and time reduction for maintenance is the most desirable factor for maintenance personnel and this is provided by modern diagnostic techniques.

In this project, the author presented the eXpress software for the process of automatically computing a diagnosis by building a dependency model of fuel-system, based upon a UAV fuel system test rig. The diagnostic analysis allowed the system to automatically compute the testability of the system. To alleviate the computational complexity of the diagnostic analysis, the software built fault detection and isolation algorithm is introduced into the eXpress software. These algorithms allow the testability of the model to a form which facilitates faster automated fault detection and identification. The various test set procedures, cost and time effective isolation results are discussed here; provides the user the benefits of eXpress software in the prognosis and diagnosis world.

As per the testability analysis three different approach methods are taken into account. These three methods provide a solid approach of testing the model with various tests. The model provides fault detection and isolation values for each tests indicates that the model works perfectly. The different algorithms provided by the software helps the user in selecting the algorithm based upon their needs. This adds an advantage for the present modern methodology in reducing the work applies to the testability method. The algorithm selected in this thesis as shown in the Tab. 4-2 is well suited for this model; the reports produced due to this model also resemble the same.

The tests defined for this model such as filter inspection test, pump operation test, shutoff valve operation test and signature test give their maximum fault detection and isolation statistics. Hence the implementation on the fuel system test rig of this model is a health management viable solution.

The cost and time factor prediction through multiple failure isolation report is most helpful to the designers and maintenance personnel. Two different approaches are carried out on this, one approach reflects the positive differential which means profitability to the user and the other approach reflects the negative differential which explains loss to the customer. The values generated on this report make the user to find the reliability, cost and time values of a particular component suitable to the model to get the maximum benefit.

The Java applet presents a web interface which easy access of all eXpress design files. It helps the designers and maintenance personnel's for easy sharing of data and accessing the file in any web browser. This will add an advantage of easy access to the modern testability designers to share the file and transfer the models easily and effectively.

In the future, this project will be carried out to the next phase by exporting this model into Diagnostic Modelling Language (DiagML) and design the file suitable to insert a model based reasoner then implement it into the fuel system test rig.

REFERENCES

[1] Dr. Steve Hobbs, *Introduction to Integrated Vehicle Health Management Technologies*, 31 March- 2 April 2009, Cranfield University.

[2] G Vachtsevanos, F Lewis, M Roeme (2006), Intelligent Fault Diagonis & Prognosis for Engineering System, John Wiley & Sons.

[3] Dr. Ashok Srivastava Ph.D., Claudia Meyer., Robert W. Mah, Ph.D., NASA, "Integrated Vehicle Health Management Technical Plan, Version 2.0." 14th April 2008.

[4] Dr. Ashok Srivastava., NASA Aviation Safety Program,"Integrated Vehicle Health Management Technical Plan Summary." 2006.

[5] Industry Canada, "*Aircraft System Diagnostics, Prognostics and Health Management Technology Insight Document." (Version 0.2),* December 16, 2004; http://www.dphmcanada.org/english/view.asp?x=17.

[6] Avionics Magazine, *A special Report on Aviation Maintenance,* Avialable at http://www.aviationtoday.com/Assets/Honeywellsmall.pdf
(Accessed 21st June 2011)

[7] LIST lab. "*Commercial use of UAVs, Brief History."* Available at http://www.list.ufl.edu/uav/UAVHstry.htm (Accessed 25th July 2011)

[8] Niu Liang, "*Aircraft fuel System and Health Management.*"MSc Thesis, January 2009.

[9] Eric Gould, "*Modelling it both ways: Hybrid Diagnostic Modelling and its application to hierarchical system designs.*" DSI International., IEEE 2004.

[10] "*A Short History of Diagnostic Modelling*." Available at http://www.testability.com

[11] "*STAT User's Group'94*." STAT Newsletter, Vol.4, No.3, December 1994, pp. 4-5.

[12] Ian Moir, Allan Seabridge, "*Aircraft Systems*." Mechanical, electrical, and avionics subsystem integration (Third Edition), John Wiley & Sons,Ltd (2008).

[13] *"Benefits & Values of ISDD; a Unified System Design Process."* Industry Presentation, DSI International, 2010.

[14] "*eXpress Quick Start Guide."* DSI-EMQ, DSI International, November 2003.

[15] "eXpress Online Help" eXpress software, Version 5.11, DSI International, 2011

[16] "*eXpress Java Applet: Browser based sharing of eXpress Diagnostic Analysis*." DSI International, available at

http://www.dsiintl.com/Resources/Brochures/JavaAppletBrochure.pdf

APPENDICES

Appendix A Design Report

A.1 Bill of Materials

Bill of Materials

##	Item	Description
1	FILTER	**Which filters all the dust and water particles in the fuel**
2	MAIN TANK	**First stage of the system which carries fuel and supply it to the system**
3	NOZZLE	**Control the characteristics of the Fuel Flow especially Increases the velocity**
4	PIPE 01	**Pipe which carries fuel**
5	PIPE 02	**Pipe which carries fuel**
6	PIPE 03	**Pipe which carries fuel**
7	PIPE 04	**Pipe which carries fuel**
8	SHUTOFF VALVE	**Valve which cut off or on the fuel flow from one system to another**
9	SUMP TANK	**Last stage of the system in which the fuel is collected through this tank**
10	XTR GEAR PUMP	**Pneumatic Const Motor 2 is used for this and it boosts the fuel to required pressure and flow rate**

A.2 Basic Design Statistics

Size Classification: **Very Small (less than 100 functions)**

Objects: **10**
Components: **10**
Assemblies: **0**

Annotations: **5**
Interpretation) **1**

Interpretation) **1**
I/O Flags: **5**
Input Flags: **0**
Output Flags: **3**
Bidirectional Flags: **2**

Functions: **3 / 13**
Reference Designators / Items: **0 / 10**
Obj; by Item): **16**

Inputs: **6 / 10**
Nets / Active/Passive Propagation: **14 / 3**
Signal Definitions: **0**
Obj; by Item): **13**

Object States (by Func; by Obj; by Item): **10**
Unused Object States (by Obj): **3**
Cause; by Object): **10**

Cause): **10**
Design States: **0**

Descriptions): **3**
Operating Modes: **1**
Subsets (by Obj; by Func): **2**
Duty Cycles Lists (by Obj; by Func): **1**

Tests / Test Sets:
Operational Tests:
User-Initiated Tests:
Probe Tests:
Signature Tests: (by

Inspection Tests: (by

Group Tests:
Hierarchical Tests:

Functions:
Active / Passive Net

Output Functions (by

Unfiltered / Filtered

Failure Modes (by Func; by

Failure Effects:
Object Effects (by

Design Effects (by

Attribute Definitions (

Topological Dependencies:
Test Dependencies:

A.3 Study Statistics

Unique Designs / Design Instances:	**1 / 1**
Maximum Depth of Expanded Hierarchy:	**1**
Objects:	**10**
Components:	**10**
Connectors:	**0**
Parts:	**0**
Assemblies:	**0**
Object States:	**0**
System-Level I/O Flags:	**5**
Input Flags:	**0**
Functions:	**3 / 13**
Output Flags:	**3**
Bidirectional Flags:	**2**
Lower-Level I/O Flags:	**0**
Linked Flags:	**0**
Unlinked / Partially-Linked Flags:	**0 / 0**
Reference Designators / Items:	**0 / 10**
Nets / Signals:	**14 / 1**

Size Classification: **Tiny (less than 200 functions)**

Candidate Tests / Test Sets:	**4 / 1**
Operational Tests:	**2**
User-Initiated Tests:	**0**
Probe Tests:	**0**
Signature Tests:	**1**
Inspection Tests:	**1**
Group Tests:	**0**
Hierarchical Tests:	**0**
Test Efficiency:	**N/A**
Functions:	**46**
Active / Passive Net	
Used Output Functions:	**14**
Unused Output Functions:	**0**
Input Functions:	**16**
Failure Modes:	**13**
Used Failure Modes:	**13**
Unused Failure Modes:	**0**
Hierarchical Operating Modes:	**0**
Topological Dependencies:	**47**
Test Dependencies:	**26**

A.4 Diagnostic Settings Report

General Settings

Title: **IVHM on UAV Fuel System Test RIG**

Scope: **Entire Design**

Mode: **Normal**

Hierarchy: **Single Level**

Fault Detection Settings

Algorithm: **Detect Malfunctions with Fewest Tests**

Test Candidates: [3]:

Outputs
Internals
Fuel System Tests [D:/eXpress work MSc/IVHM17.1.exd]

Detection Weightings [8]:

1. Priority: **80**
   Entity: **Failure Probability**
   Domain: **Suspect Functions Detected**
   Type: **Sum**
   Best Equals: **Highest**

2. Priority: **60**
   Entity: **Failure Probability**
   Domain: **Suspect Failure Modes Detected**
   Type: **Sum**
   Best Equals: **Highest**

3. Priority: **40**
   Entity: **Failure Probability**
   Domain: **Suspect Functions Proven**
   Type: **Sum**
   Best Equals: **Highest**

4. Priority: **30**
   Entity: **Failure Probability**
   Domain: **Suspect Failure Modes Proven**
   Type: **Sum**
   Best Equals: **Highest**

5. Priority: **20**
   Entity: **Number of Functions**
   Domain: **Suspect Functions Detected**
   Type: **Count**
   Best Equals: **Highest**

6. Priority: **15**
   Entity: **Number of Failure Modes**
   Domain: **Suspect Failure Modes Detected**

Type: **Count**
Best Equals: **Highest**

7. Priority: **10**
   Entity: **Number of Functions**
   Domain: **Suspect Functions Proven**
   Type: **Count**
   Best Equals: **Highest**

8. Priority: **5**
   Entity: **Number of Failure Modes**
   Domain: **Suspect Failure Modes Proven**
   Type: **Count**
   Best Equals: **Highest**

Detection Cutoffs [0]:

Fault Isolation Settings

Algorithm: **Multiple-Fault: Half-Split Failure Probs. (refinement postponed)**

Test Candidates: [3]:

Outputs
Internals
Fuel System Tests [D:/eXpress work MSc/IVHM17.1.exd]

Isolation Weightings [6]:

1. Priority: **50**
   Entity: **Failure Probability**
   Domain: **Suspect Functions Detected**
   Type: **Sum**
   Best Equals: **Half-Split**

2. Priority: **50**
   Entity: **Failure Probability**
   Domain: **Suspect Functions Proven**
   Type: **Sum**
   Best Equals: **Half-Split**

3. Priority: **40**
   Entity: **Failure Probability**
   Domain: **Suspect Failure Modes Detected**
   Type: **Sum**
   Best Equals: **Half-Split**

4. Priority: **40**
   Entity: **Failure Probability**
   Domain: **Suspect Failure Modes Proven**
   Type: **Sum**
   Best Equals: **Half-Split**

5. Priority: **20**
   Entity: **Number of Items**
   Domain: **Suspect Functions Detected**
   Type: **Count**

Best Equals: **Half-Split**

6. Priority: **20**
   Entity: **Number of Items**
   Domain: **Suspect Functions Proven**
   Type: **Count**
   Best Equals: **Half-Split**

Isolation Cutoffs [2]:

1. Entity: **Number of Tests**
   Type: **Count**
   Domain: **Isolation Path**
   Modifier 1: **Test Usage = Refinement**
   Modifier 2: **None**
   Condition: **>=3**
   Mode: **Cutoff**
   Action: **Ignore in Sequence**

2. Entity: **Number of Items**
   Type: **Count**
   Domain: **Suspected Items**
   Modifier 1: **None**
   Modifier 2: **None**
   Condition: **<=1**
   Mode: **Cutoff**
   Action: **Terminate Sequence**

A.5 Aggregate Reliability Report

Summary

Aggregate Failure Rate: **136.895350**

Failure Rate Roll-up

Items	Failure Rate
Entire Design	136.895350
FILTER	22.815892
FILTER-FLTR DRV	**11.407946**
FILTER-FLTR OUT	**11.407946**
MAIN TANK	11.407946
MAIN TANK-MN TNK OUT	**5.703973**
MAIN TANK-VENT PORT 1	**5.703973**
NOZZLE	11.407946
NOZZLE	**11.407946**
PIPE 01	11.407946
PIPE 01	**11.407946**
PIPE 02	11.407946
PIPE 02	**11.407946**
PIPE 03	11.407946
PIPE 03	**11.407946**
PIPE 04	11.407946
PIPE 04	**11.407946**
SHUTOFF VALVE	11.407946
SHUTOFF VALVE-S/O VLV DRV	**5.703973**
SHUTOFF VALVE-S/O VLV OUT	**5.703973**
SUMP TANK	11.407946
SUMP TANK	**11.407946**
XTR GEAR PUMP	22.815892
XTR GEAR PUMP-PMP FLW OUT	**11.407946**
XTR GEAR PUMP-XTR PMP DRV	**11.407946**

A.6 Detection Order Report

Summary

Total Functions Detected:	**100.00%**
Total Probability Detected:	**100.00%**
Aggregate Failure Rate:	**136.895350**
Mean Time Between Failures (MTBF):	**7304.85 hours**
Mean Time to First Failure (Initial MTTF):	**37390.17 hours**

Detection Order

Detection Test			Functions Detected		Probability Detected		Prob. Detected (w/ Interference)	
##	Test Name	Type	%	Cum %	%	Cum %	%	Cum %
1	**signature test**	**Signature**	92.86	**92.86**	95.83	**95.83**	95.83	**95.83**
2	**VENT FLAG 1**	**Output Flag**	7.14	**100.00**	4.17	**100.00**	4.17	**100.00**

A.7 Detection Coverage Report

Detection Test Coverage

1. **signature test**:

Coverage [10 items / 10 functions]			
Item / Output Function(s)	Replacement Cost	Replacement Time	Failure Probability
XTR GEAR PUMP	90.00	5400.00	
XTR GEAR PUMP-PMP FLW OUT			0.166667
FILTER	90.00	5400.00	
FILTER-FLTR OUT			0.166667
NOZZLE	90.00	5400.00	
NOZZLE			0.083333
PIPE 04	20.00	1800.00	
PIPE 04			0.083333
PIPE 03	20.00	1800.00	
PIPE 03			0.083333
PIPE 02	20.00	1800.00	
PIPE 02			0.083333
PIPE 01	20.00	1800.00	
PIPE 01			0.083333
SUMP TANK	25.00	1800.00	
SUMP TANK			0.083333
SHUTOFF VALVE	90.00	5400.00	
SHUTOFF VALVE-S/O VLV OUT			0.083333
MAIN TANK	100.00	6000.00	
MAIN TANK-MN TNK OUT			0.041667
TOTALS:	565.00	36600.00	0.958333

2. **VENT FLAG 1**:

Coverage [1 item / 1 function]			
Item / Output Function(s)	Replacement Cost	Replacement Time	Failure Probability
MAIN TANK	100.00	6000.00	
MAIN TANK-VENT PORT 1			0.041667
Stimuli [1]			
VENT FLAG 1			

A.8 Item Detection Report

Item / Function(s)	Test (Test Set) [detection test #]
FILTER	
FILTER-FLTR DRV	Proven: **<not proven>** Detected: **<not detected>**
FILTER-FLTR OUT	Proven: **signature test (Fuel System Tests) [1]** Detected: **signature test (Fuel System Tests) [1]**
MAIN TANK	
MAIN TANK-MN TNK OUT	Proven: **signature test (Fuel System Tests) [1]** Detected: **signature test (Fuel System Tests) [1]**
MAIN TANK-VENT PORT 1	Proven: **VENT FLAG 1 [2]** Detected: **VENT FLAG 1 [2]**
NOZZLE	
NOZZLE	Proven: **signature test (Fuel System Tests) [1]** Detected: **signature test (Fuel System Tests) [1]**
PIPE 01	
PIPE 01	Proven: **signature test (Fuel System Tests) [1]** Detected: **signature test (Fuel System Tests) [1]**
PIPE 02	
PIPE 02	Proven: **signature test (Fuel System Tests) [1]** Detected: **signature test (Fuel System Tests) [1]**
PIPE 03	
PIPE 03	Proven: **signature test (Fuel System Tests) [1]** Detected: **signature test (Fuel System Tests) [1]**
PIPE 04	
PIPE 04	Proven: **signature test (Fuel System Tests) [1]** Detected: **signature test (Fuel System Tests) [1]**
SHUTOFF VALVE	
SHUTOFF VALVE-S/O VLV DRV	Proven: **<not proven>** Detected: **<not detected>**
SHUTOFF VALVE-S/O VLV OUT	Proven: **signature test (Fuel System Tests) [1]** Detected: **signature test (Fuel System Tests) [1]**
SUMP TANK	
SUMP TANK	Proven: **signature test (Fuel System Tests) [1]** Detected: **signature test (Fuel System Tests) [1]**
XTR GEAR PUMP	
XTR GEAR PUMP-PMP FLW OUT	Proven: **signature test (Fuel System Tests) [1]** Detected: **signature test (Fuel System Tests) [1]**
XTR GEAR PUMP-XTR PMP DRV	Proven: **<not proven>** Detected: **<not detected>**

A.9 Fault Isolation Report

Multiple Failures
Fault Group Size Statistics

	Isolation Percentages Using Testing Only			Isolation Probabilities Using Testing Only		Resolution Probabilities Using Lambda Search	
Size	Qty	%	Cum %	%	Cum %	%	Cum %
1	11	100.00	**100.00**	100.00	**100.00**	100.00	**100.00**

Total Fault Groups:	**11**
Average Fault Group Size:	**1.00**
Isolation Effectiveness:	**100.00**
Expected Fault Group Size:	**1.00**
Resolution Effectiveness:	**100.00**
Expected Repairs per Isolation:	**1.00**

Fault Group Counting Method: Groups containing the same **functions** are counted as the same group
Lambda Search: Max. of **25** items, where the comparison with the **next highest item** is at least **1.00 to 1**

Cost/Time to Diagnose a Primary Failure
Using Testing Only (Multiple Failure Isolation)

Cost to Isolate (in US Dollars)			**Time to Isolate** (in minutes)		
Minimum:	**2.00**		Minimum:	**1800.00**	
Maximum:	**2.00**		Maximum:	**1800.00**	
Average:	**2.00**		Average:	**1800.00**	
Expected (MCTI):	**2.00**		Expected (MTTI):	**1800.00**	
Differential:	**0.00**	**(0.00%)**	Differential:	**0.00**	**(0.00%)**
per Operating Hour:	**0.0003**		per Operating Hour:	**0.2464**	
Cost to Replace (in US Dollars)			**Time to Replace** (in minutes)		
Minimum:	**20.00**		Minimum:	**1800.00**	
Maximum:	**100.00**		Maximum:	**6000.00**	
Average:	**60.45**		Average:	**3872.73**	
Expected:	**62.08**		Expected:	**3950.00**	
Differential:	**1.63**	**(+2.69%)**	Differential:	**77.27**	**(+2.00%)**
per Operating Hour:	**0.0085**		per Operating Hour:	**0.5407**	
Cost to Repair (in US Dollars)			**Time to Repair** (in minutes)		
Minimum:	**22.00**		Minimum:	**3600.00**	
Maximum:	**102.00**		Maximum:	**7800.00**	
Average:	**62.45**		Average:	**5672.73**	
Expected (MCTR):	**64.08**		Expected (MTTR):	**5750.00**	
Differential:	**1.63**	**(+2.61%)**	Differential:	**77.27**	**(+1.36%)**
per Operating Hour:	**0.0088**		per Operating Hour:	**0.7871**	

Mean Time Between Failures (MTBF): **7304.85 hours**
Inherent Availability: **0.987051**

A.10 Fault Group Statistics

Multiple Failure
Fault Group Details

Fault Group # 0			
Item [1] / Function [1]	Replacement Cost	Replacement Time	Failure Probability
MAIN TANK	100.00	6000.00	
cc MAIN TANK-MN TNK OUT			0.041667

Fault Group # 1			
Item [1] / Function [1]	Replacement Cost	Replacement Time	Failure Probability
PIPE 01	20.00	1800.00	
cc PIPE 01			0.083333

Fault Group # 2			
Item [1] / Function [1]	Replacement Cost	Replacement Time	Failure Probability
FILTER	90.00	5400.00	
cc FILTER-FLTR OUT			0.166667

Fault Group # 3			
Item [1] / Function [1]	Replacement Cost	Replacement Time	Failure Probability
PIPE 02	20.00	1800.00	
cc PIPE 02			0.083333

Fault Group # 4			
Item [1] / Function [1]	Replacement Cost	Replacement Time	Failure Probability
XTR GEAR PUMP	90.00	5400.00	
cc XTR GEAR PUMP-PMP FLW OUT			0.166667

Fault Group # 5			
Item [1] / Function [1]	Replacement Cost	Replacement Time	Failure Probability
PIPE 03	20.00	1800.00	
cc PIPE 03			0.083333

Fault Group # 6			
Item [1] / Function [1]	Replacement Cost	Replacement Time	Failure Probability
SHUTOFF VALVE	90.00	5400.00	

Fault Group # 7			
Item [1] / Function [1]	Replacement Cost	Replacement Time	Failure Probability
PIPE 04	20.00	1800.00	
cc PIPE 04			0.083333
cc SHUTOFF VALVE-S/O VLV OUT			0.083333

Fault Group # 8			
Item [1] / Function [1]	Replacement Cost	Replacement Time	Failure Probability
NOZZLE	90.00	5400.00	
cc NOZZLE			0.083333

Fault Group # 9			
Item [1] / Function [1]	Replacement Cost	Replacement Time	Failure Probability
SUMP TANK	25.00	1800.00	
cc SUMP TANK			0.083333

Fault Group # 10			
Item [1] / Function [1]	Replacement Cost	Replacement Time	Failure Probability
MAIN TANK	100.00	6000.00	
cc MAIN TANK-VENT PORT 1			0.041667

A.11 Test Point Placement Report

Test Location Rankings				
Location	Type	## of Tests	Pctg. of Isolations	Usage Probability
VENT FLAG 2	IO Flag	1	100.00	100.00
PMP FLW	Net	1	90.91	95.83
FTR FLW	Net	1	45.45	54.17
S/O VLV FLW 1	Net	1	45.45	41.67
MT FLW 1	Net	1	27.27	29.17
S/O VLV FLW	Net	1	27.27	25.00
FTR FLW 2	Net	1	18.18	25.00
NZL FLW	Net	1	18.18	16.67
PMP FLW 1	Net	1	18.18	16.67
MT FLW	Net	1	18.18	12.50
VENT FLAG 1	IO Flag	1	9.09	4.17

Test Rankings				
Test	Type	## of Nodes	Pctg. of Isolations	Usage Probability
signature test	Signature	1	100.00	100.00
VENT FLAG 1 [XTR GEAR PUMP-PMP FLW OUT]	Internal Signal	1	90.91	95.83
VENT FLAG 1 [FILTER-FLTR OUT]	Internal Signal	1	45.45	54.17
VENT FLAG 1 [PIPE 04-S/O VLV FLW 1]	Internal Signal	1	45.45	41.67
VENT FLAG 1 [PIPE 01-MT FLW 1]	Internal Signal	1	27.27	29.17
VENT FLAG 1 [SHUTOFF VALVE-S/O VLV OUT]	Internal Signal	1	27.27	25.00
VENT FLAG 1 [PIPE 02-FTR FLW 2]	Internal Signal	1	18.18	25.00
VENT FLAG 1 [NOZZLE-NZL FLW OUT]	Internal Signal	1	18.18	16.67
VENT FLAG 1 [PIPE 03-PMP FLW 1]	Internal Signal	1	18.18	16.67
VENT FLAG 1 [MAIN TANK-MN TNK OUT]	Internal Signal	1	18.18	12.50
VENT FLAG 1	Output Flag	1	9.09	4.17

Test Node Rankings				
Test Node	Test	Type	Pctg. of Isolations	Usage Probability
1-0	signature test	Detection	100.00	100.00
1-1	VENT FLAG 1 [XTR GEAR PUMP-PMP FLW OUT]	Isolation	90.91	95.83
1-2	VENT FLAG 1 [FILTER-FLTR OUT]	Isolation	45.45	54.17
1-6	VENT FLAG 1 [PIPE 04-S/O VLV FLW 1]	Isolation	45.45	41.67
1-3	VENT FLAG 1 [PIPE 01-MT FLW 1]	Isolation	27.27	29.17
1-7	VENT FLAG 1 [SHUTOFF VALVE-S/O VLV OUT]	Isolation	27.27	25.00
1-5	VENT FLAG 1 [PIPE 02-FTR FLW 2]	Isolation	18.18	25.00
1-8	VENT FLAG 1 [PIPE 03-PMP FLW 1]	Isolation	18.18	16.67
1-9	VENT FLAG 1 [NOZZLE-NZL FLW OUT]	Isolation	18.18	16.67
1-4	VENT FLAG 1 [MAIN TANK-MN TNK OUT]	Isolation	18.18	12.50
2-0	VENT FLAG 1	Detection	9.09	4.17

A.12 FD/FI Statistics by Category Report

Fault Detection by Category			
Category	Percentage Detected in Category	Probability Detected in Category	Overall Probability Detected
MAIN TANK FEED	**100.00**	**100.00**	**91.67**

Fault Isolation by Category									
Category	Fault Isolation Percentages (Testing only)			Fault Isolation Probabilities (Testing only)			Fault Resolution Probabilities (Lambda Search)		
	1	2	3	1	2	3	1	2	3
MAIN TANK FEED	**100.00**			**100.00**			**100.00**		

Lambda Search Criteria: maximum of **1** replacement, where the ratio with the **next highest item** must exceed **100.00 to 1**

A.13 Diagnostic Flow Table

Record	Test / (Test Set)	Previous Record	Decision Path		Goto Record
1-0	signature test (Fuel System Tests)		FOUND: N/F:	VENT FLAG 1 VENT FLAG 1 [XTR GEAR PUMP-PMP FLW OUT]	2-0 1-1
1-1	VENT FLAG 1 [XTR GEAR PUMP-PMP FLW OUT]	1-0	GOOD: BAD:	VENT FLAG 1 [PIPE 04-S/O VLV FLW 1] VENT FLAG 1 [FILTER-FLTR OUT]	1-6 1-2
1-2	VENT FLAG 1 [FILTER-FLTR OUT]	1-1	GOOD: BAD:	VENT FLAG 1 [PIPE 02-FTR FLW 2] VENT FLAG 1 [PIPE 01-MT FLW 1]	1-5 1-3
1-3	VENT FLAG 1 [PIPE 01-MT FLW 1]	1-2	GOOD: BAD:	**Isolated Fault Group # 2 (1 Item):** FILTER VENT FLAG 1 [MAIN TANK-MN TNK OUT]	 1-4
1-4	VENT FLAG 1 [MAIN TANK-MN TNK OUT]	1-3	GOOD: BAD:	**Isolated Fault Group # 1 (1 Item):** PIPE 01 **Isolated Fault Group # 0 (1 Item):** MAIN TANK	
1-5	VENT FLAG 1 [PIPE 02-FTR FLW 2]	1-2	GOOD: BAD:	**Isolated Fault Group # 4 (1 Item):** XTR GEAR PUMP **Isolated Fault Group # 3 (1 Item):** PIPE 02	
1-6	VENT FLAG 1 [PIPE 04-S/O VLV FLW 1]	1-1	GOOD: BAD:	VENT FLAG 1 [NOZZLE-NZL FLW OUT] VENT FLAG 1 [SHUTOFF VALVE-S/O VLV OUT]	1-9 1-7
1-7	VENT FLAG 1 [SHUTOFF VALVE-S/O VLV OUT]	1-6	GOOD: BAD:	**Isolated Fault Group # 7 (1 Item):** PIPE 04 VENT FLAG 1 [PIPE 03-PMP FLW 1]	 1-8
1-8	VENT FLAG 1 [PIPE 03-PMP FLW 1]	1-7	GOOD: BAD:	**Isolated Fault Group # 6 (1 Item):** SHUTOFF VALVE **Isolated Fault Group # 5 (1 Item):** PIPE 03	
1-9	VENT FLAG 1 [NOZZLE-NZL FLW OUT]	1-6	GOOD: BAD:	**Isolated Fault Group # 9 (1 Item):** SUMP TANK **Isolated Fault Group # 8 (1 Item):** NOZZLE	
2-0	VENT FLAG 1	1-0	GOOD: BAD:	**No Faults Encountered** **Isolated Fault Group # 10 (1 Item):** MAIN TANK	

A.14 Fault Isolation Report

Multiple Failure
Fault Group Size Statistics

	Isolation Percentages Using Testing Only			Isolation Probabilities Using Testing Only		Resolution Probabilities Using Lambda Search	
Size	Qty	%	Cum %	%	Cum %	%	Cum %
1	11	100.00	**100.00**	100.00	**100.00**	100.00	**100.00**

Total Fault Groups:	**11**
Average Fault Group Size:	**1.00**
Isolation Effectiveness:	**100.00**
Expected Fault Group Size:	**1.00**
Resolution Effectiveness:	**100.00**
Expected Repairs per Isolation:	**1.00**

Fault Group Counting Method: Groups containing the same **functions** are counted as the same group
Lambda Search: Max. of **25** items, where the comparison with the **next highest item** is at least **1.00 to 1**

Cost/Time to Diagnose a Primary Failure
Using Testing Only (Multiple Failure Isolation)

Cost to Isolate (in US Dollars)		**Time to Isolate** (in minutes)	
Minimum:	**2.00**	Minimum:	**1800.00**
Maximum:	**2.00**	Maximum:	**1800.00**
Average:	**2.00**	Average:	**1800.00**
Expected (MCTI):	**2.00**	Expected (MTTI):	**1800.00**
Differential:	**0.00 (+0.00%)**	Differential:	**0.00 (+0.00%)**
per Operating Hour:	**0.0003**	per Operating Hour:	**0.2347**
Cost to Replace (in US Dollars)		**Time to Replace** (in minutes)	
Minimum:	**10.00**	Minimum:	**1200.00**
Maximum:	**200.00**	Maximum:	**9000.00**
Average:	**66.36**	Average:	**3763.64**
Expected:	**50.06**	Expected:	**2943.75**
Differential:	**-16.30 (-24.56%)**	Differential:	**-819.89 (-21.78%)**
per Operating Hour:	**0.0065**	per Operating Hour:	**0.3838**
Cost to Repair (in US Dollars)		**Time to Repair** (in minutes)	
Minimum:	**12.00**	Minimum:	**3000.00**
Maximum:	**202.00**	Maximum:	**10800.00**
Average:	**68.36**	Average:	**5563.64**
Expected (MCTR):	**52.06**	Expected (MTTR):	**4743.75**
Differential:	**-16.30 (-23.84%)**	Differential:	**-819.89 (-14.74%)**
per Operating Hour:	**0.0068**	per Operating Hour:	**0.6185**

Mean Time Between Failures (MTBF):	**7670.09 hours**
Inherent Availability:	**0.989797**

Appendix B Diagnosis Algorithm

B.1 Detection Algorithm

1) The Detect Malfunctions with Fewest Tests fault detection algorithm attempts to provide the most direct path to a detected failure. Assuming that the desire to detect failures using the fewest tests implies a certain sense of expediency, the Test Candidate Groupings for this algorithm dictate that all intrusive tests be postponed until all useful non-intrusive tests have been performed. Within these groupings, the Test Weightings are set up so that the test that is most likely to fail is selected first. In other words, the first test in the detection order will be the one that is most likely to fail (that is, most likely to detect a failure). The second test will be that which is most likely to fail, given that the first test did not. And so on.

2) The fault detection algorithm Detect Probable Malfunctions is similar to the algorithm Detect Malfunction with Fewest Tests. Both algorithms, for example, privilege tests that are more likely to detect a failure (in fact, the two algorithms use identical Test Weightings). The main difference is that the non-intrusive tests, although still performed before intrusive tests, are now split into two different Test Candidate Groupings. In this algorithm, non-intrusive tests that are located at internal output flags are performed before tests located at terminal output flags. Although diagnostics generated using this algorithm will tend to require more tests to detect a failure than those developed using the algorithm Detect Malfunction with Fewest Tests, the diagnostics for this algorithm will require less fault isolation (since better isolation is achieved during detection). Furthermore, this algorithm makes better use of all defined test points (the other algorithm will ignore tests defined at internal output flags when their functional coverage is also associated with tests defined at terminal output flags). For both algorithms, however, the prioritization of instrusive tests is the same.

3) The Detect Critical Malfunctions fault detection algorithm attempts to provide the most direct path to a critical failure. The first eight Test Weightings are set up so that the tests that are most likely to detect the most severe failures (that is, the failures with the highest end item effect severity) are selected first. The second eight Test Weightings are identical to those defined for the algorithm Detect Malfunctions with Fewest Tests, as are the Test Candidate Groupings. This means that, if two tests cover equally critical failures, the test that is most likely to fail is selected first. Also, if failure effects are not defined for the given design (or if all failures are equally severe), this algorithm will produce the same detection order as the algorithm Detection Malfunctions with Fewest Tests.

4) The Prove Operation with Fewest Tests fault detection algorithm attempts to verify operational integrity using the smallest number of tests. Assuming that the desire to minimize testing implies a certain sense of expediency, the Test Candidate Groupings for this algorithm dictate that all intrusive tests be postponed until all useful non-intrusive tests have been performed. In fact, the groupings for this algorithm are the same as those defined for the algorithm Detect Malfunction with Fewest Tests. Within these groupings, the Test Weightings are set up so that the test that proves the most will be selected first. In other words, the first test in the detection order will be the one that exonerates the set of functions or failure modes with the largest cumulative failure probability. The second detection test selected will be that which proves the most, given that the first test did not fail. And so on.

5) The Prove Maximum Operation Before Detecting Malfunction fault detection algorithm attempts to prove good as much of the design as possible before detecting a failure (so that, if multiple-fault isolation is used, better isolation can be achieved once a fault is detected). Unlike many of the other detection algorithms, this algorithm does not distinguish between intrusive and non-intrusive tests (although it does use Test Candidate Groupings to postpone

Probe Tests until all other types of tests have been performed). The assumption here is that the analyst is willing to perform additional or less convenient testing in exchange for better fault isolation.

6) The Minimize Switches in Monitored Stimuli fault detection algorithm attempts to reduce the wear and tear on test equipment by reducing the number of input configurations required during the fault detection process. It does this using two Test Weightings that minimize the differences and maximize the similarities between the stimuli locations that are monitored for each test. This feature can be incorporated into other diagnostic algorithms (since this is the only predefined algorithm that attempts to minimize stimuli switches) by simply replicated the first two weightings from this algorithm.

7) The Detect Using Fault Codes algorithm is perhaps the least complex of the diagnostic algorithms that have been predefined for fault detection, with only two Test Candidate Groupings ("Test Set Tests" and "Output Flags"). This algorithm has been specially designed to utilize tests that can usually detect a failure when it occurs, but cannot necessarily rule out that failure if it is not observed. This is thus the only predefined detection algorithm that will perform tests that prove nothing when they pass (e.g. Signature Identifies Malfunction or Inspect for Malfunction tests). Although the Test Weightings for this algorithm will result in the test being performed first that can detect failures associated with the most items, output functions or failure modes, it must be remembered that lists of detection tests created using this algorithm could become relatively long (since failures not exonerated may be tested more than once).

B.2 Isolation Algorithm

1) The Multiple Fault: Half-Split Failure Probs. (refinement postponed) fault isolation algorithm effects what is essentially a compromise between the size of the generated test sequence and the facility with which that sequence is able to isolate. In order to reduce the total number of nodes in the generated test sequence, this Multiple-Failure algorithm always performs tests that are guaranteed to reduce the size of the suspect set (regardless of whether the test

passes or fails) before tests that only reduce the suspect set if that test passes or if it fails (one or the other). What is sacrificed is the fact that some of the refinement tests may be much more easily performed than some of the isolation tests that this algorithm forces to precede them (for example, most signature tests and all operational tests defined at untested terminal output flags would generally be treated as refinement tests). Nevertheless, for many systems, the diagnostic test sequences that would result if refinement tests were allowed to precede isolation tests would be prohibitively large. This algorithm helps reduce the size of that sequence. One of the quirks of this algorithm is that it will perform output flags or net functions as isolation tests before it uses a test set test for refinement (this is the only predefined algorithm that allows a generic test to precede a test set test). This is because the algorithm assumes that the only time that output flags or net functions are included as tests in diagnostics would be when the analyst is attempting to determine how to improve on the isolation that can be achieved prior to refinement, since isolation tests are more efficient than refinement tests. Remember, when the final diagnostics are generated, all testing should be performed using test set tests. The weightings for this algorithm favor tests that come close to half-splitting the suspect set when they pass or fail.

2) The fault isolation algorithm Multiple Fault: Maximize Functions Proven by Refinement, like the algorithm Multiple Fault: Half-Split Failure Probs. (refine where appropriate), uses Multiple-Failure isolation and does not constrain Test Set Test selection using Test Candidate Groupings. All Test Set tests are equally candidates-the first test to be performed will be the test that best satisfies the algorithm's weighting criteria. The weightings for this algorithm attempt to reduce the number of refinement tests by first using tests that prove the maximum failure-weighted percentage of functions and/or failure modes when they pass (a secondary set of criteria will privilege the tests that come closest to half-splitting when they fail).

3) The fault isolation algorithm Multiple Fault: Half-Split Failure Probs. (no refinement) is the same as the algorithm Multiple Fault: Half-Split Failure Probs. (refinement postponed) with the exception that refinement tests are omitted altogether from the generated diagnostic test sequence. This algorithm is a good one to use when trying to improve isolation for a design. One of the other algorithms can be used to determine how good the isolation is, whereas this one can be used to determine what tests to develop in order to improve isolation. Then, once those tests are defined, the analyst can switch back to the other algorithm (which allows refinement) in order to evaluate the full benefit of the additional testing. The weightings for this algorithm favor tests that come close to half-splitting the suspect set when they pass or fail.

4) The fault isolation algorithm Multiple Fault: Static Health Monitoring (Operational refinement not postponed) is a compromise between the two algorithms Multiple Fault: Half-Split Failure Probs. (refinement postponed) and Multiple Fault: Half-Split Failure Probs. (refine where appropriate). Rather than performing the most efficient tests first (that is, isolation tests that are guaranteed to reduce the suspect set), this algorithm produces test sequences that will use Operational Tests to perform refinement prior to performing isolation tests. The reason why this is useful is that Operational Tests are often used to represent tests that are simple, quick and inexpensive to perform (they may perhaps be automatic) and that can provide instant feedback about the health of the system. On the down side, because of the inefficiency of refinement tests, test sequences generated using this algorithm can be substantially larger than those generated using the other predefined isolation algorithms (remember, for refinement tests, only one of the two possible test outcomes is diagnostically meaningful). The highest-priority weightings for this algorithm are those that favor tests that come close to half-splitting the suspect set when they pass. Lower-priority weightings attempt to choose the refinement tests that are least likely to fail due to malfunctions not in the suspect set.

5) The fault isolation algorithm Multiple Fault: Maximize Functions Proven by Refinement, like the algorithm Multiple Fault: Half-Split Failure Probs. (refine where appropriate), uses Multiple-Failure isolation and does not constrain Test Set Test selection using Test Candidate Groupings. All Test Set tests are equally candidates-the first test to be performed will be the test that best satisfies the algorithm's weighting criteria. The weightings for this algorithm attempt to reduce the number of refinement tests by first using tests that prove the maximum failure-weighted percentage of functions and/or failure modes when they pass (a secondary set of criteria will privilege the tests that come closest to half-splitting when they fail).

6) Common Cause: Half-Split Failure Probs. is one of two pre-defined fault isolation algorithms that uses Common Cause, rather than Multiple-Failure Isolation (remember, the analyst is not limited to using the predefined algorithms, but may edit them to create custom diagnostic solutions). Because this algorithm utilizes Common Cause isolation, it does not need to account for the idiosyncrasies of refinement testing. This algorithm uses only two Test Candidate Groupings: one for test set tests and another for output flags and internal net functions. This algorithm also uses fewer weightings (3) and cutoffs (1) than do the multiple fault algorithms, so diagnostic calculation may complete more quickly. The weightings for this algorithm favour tests that come close to half-splitting the suspect set when they pass.

7) The fault isolation algorithm Common Cause: Half-Split Failure Probs. (Max. Depth = 10) is the same as the algorithm Common Cause: Half-Split Failure Probs. with the added constraint that no more than 10 tests can be used in each isolation path. This constraint is implemented as an additional cutoff which prevents isolation from continuing after 10 isolation tests have been performed. The weightings for this algorithm favour tests that come close to half-splitting the suspect set when they pass.

B.3 FMECA Chart

ID	Item	Failure	Root Failure Mode Causes	Failure Effects			Compensating Provisions	Severity Class	Failure Ratio	Failure Rate
				Local	Next Higher	End Item				
1	FILTER	INABILITY TO TRANSFER FUEL FROM FLTRIN TO FLTR OUT	FM2 CLOGGED FILTER(FULLY)	INABILITY TO TRANSFER FUEL FROM FLTRIN TO FLTR OUT		INABILITY TO TRANSFER FUEL FROM FLTRIN TO FLTR OUT		Loss of Operation	80	18.252713
2		TRANSFER RATE REDUCED FROM FLTRIN TO FLTROUT	FM1 CLOGGED FILTER(PARTLY)	TRANSFER RATE REDUCED FROM FLTRIN TO FLTROUT		TRANSFER RATE REDUCED FROM FLTRIN TO FLTROUT		Degraded Performance	20	4.563178
3	NOZZLE	INABILITY TO SUPPLY FUEL COMPLETELY	CLOGGED NOZZLE FULLY	INABILITY TO SUPPLY FUEL COMPLETELY		INABILITY TO SUPPLY FUEL COMPLETELY		Loss of Operation	80	9.126357
4		INABILITY TO SUPPLY FUEL PARTLY	CLOGGED NOZZLE PARTLY	INABILITY TO SUPPLY FUEL PARTLY		INABILITY TO SUPPLY FUEL PARTLY		Degraded Performance	20	2.281589
5	PIPE 04	INABILITY TO GET FUEL COMPLETELY	LEAKING PIPE FULLY	INABILITY TO GET FUEL COMPLETELY		INABILITY TO GET FUEL COMPLETELY		Loss of Operation	80	9.126357
6		INABILITY TO GET FUEL CORRECTLY	LEAKING PIPE PARTLY	INABILITY TO GET FUEL CORRECTLY		INABILITY TO GET FUEL CORRECTLY		Degraded Performance	20	2.281589
7	SHUTOFF VALVE	INABILITY TO CONTROL FUEL FLOW	FM5 S/O VLV(STK OPEN) FM9 LEAKING (Internally)	INABILITY TO CONTROL FUEL FLOW		INABILITY TO CONTROL FUEL FLOW		Loss of Operation	50	5.703973
8		INABILITY TO TRANSFER FUEL FLOW	FM6 S/O VLV(STK CLOSED) FM7 STICKING FM8 LEAKING (Externally)	INABILITY TO TRANSFER FUEL FLOW		INABILITY TO TRANSFER FUEL FLOW		Loss of Operation	50	5.703973
9	XTR GEAR PUMP	INABILITY TO PUMP FUEL	FM3 XTR PMP(FULLY FAULT)	INABILITY TO PUMP FUEL		INABILITY TO PUMP FUEL		Loss of Operation	80	18.252713
10		REDUCED PERFORMANCE OF PMP	FM4 XTR PMP(PARTLY FAULT)	REDUCED PERFORMANCE OF PMP		REDUCED PERFORMANCE OF PUMP		Degraded Performance	20	4.563178