
Reliability Workbench V10.3 Technical Specification

IMPORTANT:

Reliability Workbench 10.3 is functionally identical to **Reliability Workbench 10.2.4**. It only differs in the method of licensing and installation. As an existing customer, unless you are experiencing problems with your current license (EG borrowing), we recommend installing 10.2.4.

An upgrade to Reliability Workbench 10.3 will require you to obtain a new license from us. If you have any queries, please email support@isograph.com

Platform

Runs under Windows 95, 98, NT, 2000, Me, Xp, Vista* and 7*. Recommended host memory requirements 128Mb+

** The application WinHlp32.exe has been used to run help files with the extension '.hlp' since Windows v3.1. Microsoft have removed it for the release of Vista and future OS's. The Reliability Workbench help files require WinHlp32.exe to run. Though the application is no longer provided with Windows, it is still supported and may be downloaded from the following link:*

<http://www.microsoft.com/downloads/details.aspx?FamilyID=6ebcfad9-d3f5-4365-8070-334cd175d4bb&DisplayLang=en>

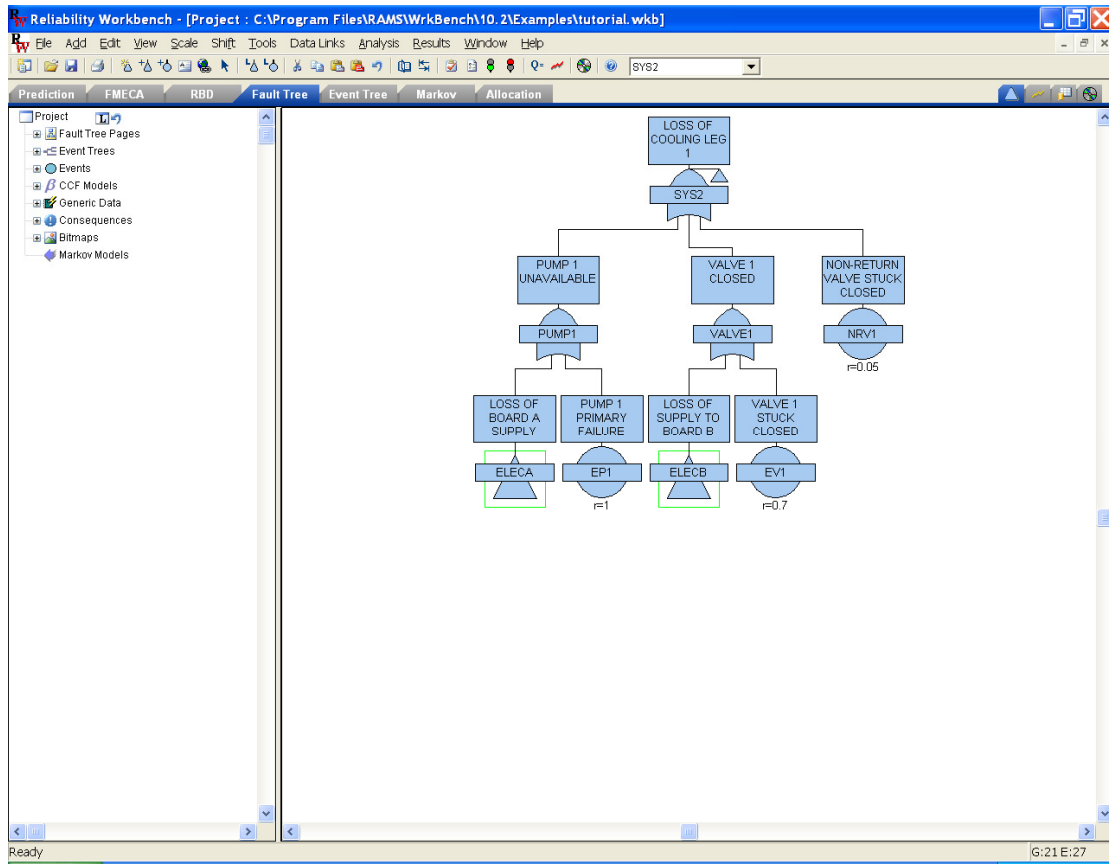
Summary of analysis capabilities :

- MIL-217F Notice 2 Prediction
 - Parts Count
 - Stress Analysis
 - Commercial Libraries
- Telcordia TR/SR-332 Prediction
 - Steady-State Failure Rate Prediction
 - First Year Multipliers
 - Integration of Lab/Field Data
- UTE C 80-810(RDF 2000) Prediction
- NSWC standard 98/LE1 Prediction
- GJB/Z 299B Prediction
- GJB/Z 299C Prediction
- 217 Plus Prediction
- RAC Prism Link
- Maintainability Prediction
- FMECA
 - MIL-STD 1629
 - Process and Design FMEA
 - Commercial Aircraft FMEA
 - Custom Formats
 - IEC 61508-6 diagnostic coverage models
- Reliability Block Diagram Analysis
 - Minimal Cut Set Analysis
 - CCF Analysis
 - Importance Analysis
 - System Quantification
 - House Event Analysis
 - IEC 61508-6 treatment of dormant failures
- Fault Tree Analysis
 - Minimal Cut Set Analysis
 - Combinatorial Cut Set Analysis
 - CCF Analysis
 - Importance Analysis
 - System Quantification
 - Confidence Analysis
 - House Event Analysis
 - IEC 61508-6 treatment of dormant failures
- Event Tree Analysis
 - Minimal Cut Set Analysis
 - CCF Analysis
 - Importance Analysis
 - Risk Analysis
- Markov Analysis
 - Numerical Integration
 - Time-dependent Analysis
 - Semi-Markov Models

Isograph Reliability Software

www.isograph.com

- Parts Libraries
 - NPRD-95
 - IAEA
- Commercial Electronic Parts



Fault Tree Module Screen Shot

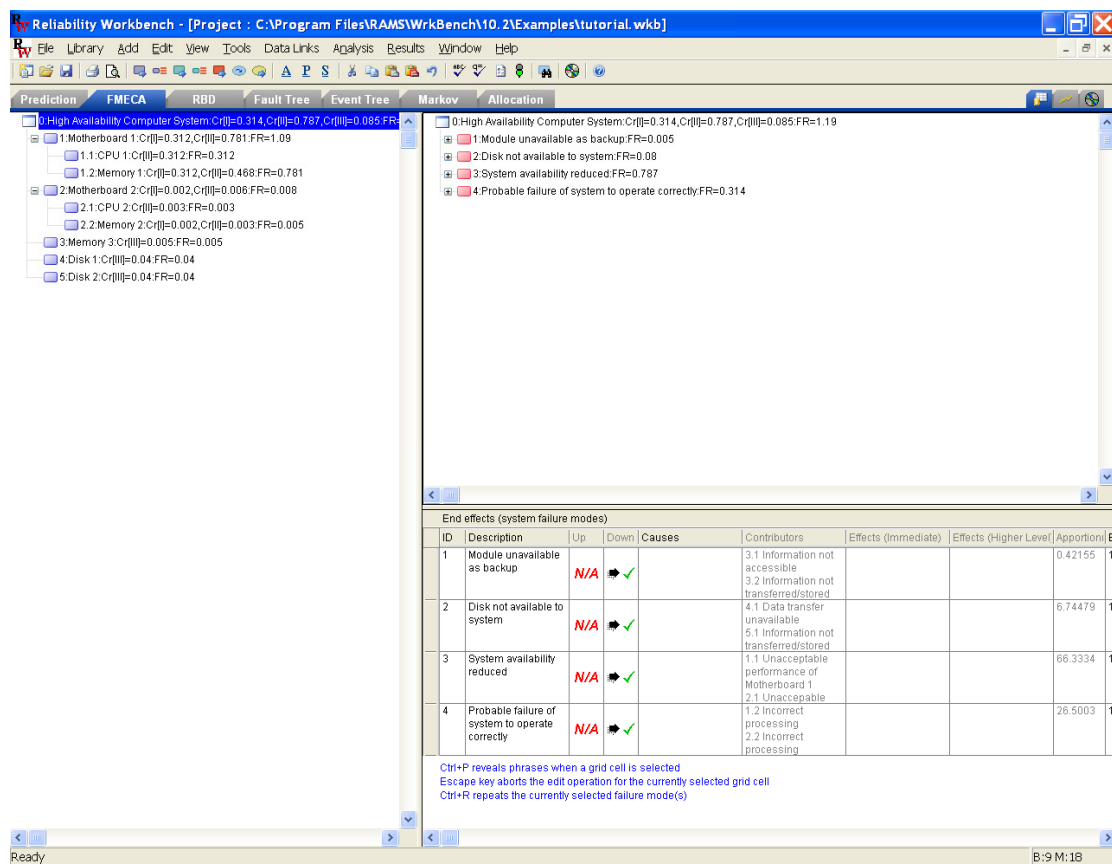
Introduction

Welcome to Reliability Workbench, the integrated environment for performing Reliability Prediction, Maintainability Prediction, Failure Mode Effect and Criticality Analysis (FMECA), Reliability Block Diagram (RBD) analysis, Fault Tree Analysis, Event Tree Analysis and Markov Analysis.

Reliability Workbench allows users to develop projects containing data appropriate to one or more of these analysis modules. Data may easily be transferred between these modules simply by using copy and paste and the automatic data transfer facilities. If you are using the MIL-217 or Telcordia prediction modules you will also be able to automatically convert data between these two different standards.

When operating the Reliability Workbench program you can switch between the various modules in an instant. Each module may be activated by selecting the appropriate tab in the edit window. Data may be transferred between modules simply by copying and pasting the appropriate data. Links between modules are maintained using a powerful data link facility and data can be automatically updated in one module due to changes made in another module. For example, the user may require that a block in a RBD obtain its failure rate from a MIL-217 prediction. The block in the RBD may therefore be linked to a block in the MIL-217 prediction module. Changes in the failure rates in the Prediction Module may then be automatically transferred through to the RBD. Another block in the RBD might obtain its failure rate data from a NSWC prediction.

Reliability Workbench runs under Windows 95, Windows 98, Windows NT, Windows 2000, Windows Me and Windows Xp. It provides a powerful Report Generator allowing users to construct customised reports and graphs. In addition a powerful import/export facility is provided allowing data to be transferred directly to and from Microsoft Access databases and spreadsheet programs such as Microsoft Excel.



FMECA Module Screen Snapshot

Prediction Module Features

The Prediction Module is a powerful reliability prediction program which includes the internationally recognised method of calculating electronic equipment reliability given in MIL-HDBK-217 (published by the US Department of Defense), RDF 2000, Telcordia Standards TR-332 Issue 6 and SR-332 Issue 1, GJB/Z 299B and 217 Plus.

These standards use a series of models for various categories of electronic, electrical and electro-mechanical components to predict failure rates which are affected by environmental conditions, quality levels, stress conditions and various other parameters. These models are fully detailed in the appropriate standards.

The Telcordia Standard allows reliability predictions to be performed using three methods. Method I provides predictions based on a *Parts Count* procedure. Method II provides predictions based on combining laboratory test data with *Parts Count* data. Method III provides predictions based on combining field tracking data with *Parts Count* data. All three methods are handled by the Prediction Module.

The Telcordia Standard also provides models for predicting the failure rates of units and devices during the first year of operation. The failure rate during this wear-in phase is expressed as a multiplying factor operating on the predicted steady-state failure rate. This *First Year Multiplier* (FYM) is influenced by burn-in times and temperatures. The Prediction Module automatically calculates the *First Year Multiplier* based on specified system, unit and device burn-in times and temperatures.

The Prediction Module also provides mechanical equipment failure rates according to the NSWC Standard 98/LE1 (Handbook of Reliability Prediction Procedures for Mechanical Equipment).

This standard uses a series of models for various categories of mechanical components to predict failure rates which are affected by temperature, stresses, flow rates and various other parameters. These models are fully detailed in NSWC 98/LE1. The categories of mechanical equipment covered by the Prediction Module are :

- Seals and Gaskets
- Springs
- Solenoids
- Valve Assemblies
- Bearings
- Gears and Splines
- Actuators
- Pumps
- Filters
- Brakes and Clutches
- Compressors
- Electric Motors
- Threaded Fasteners
- Mechanical Couplings
- Slider-Crank Mechanisms

Many of the categories of mechanical equipment are in fact composed of a collection of sub-components which must be modelled by the user. Typical collections include

- Valve Assemblies - poppet/sliding action assembly, seals, springs, solenoids, housing
- Pumps - shafts, seals, bearings, casing, fluid driver
- Brakes & Clutches - actuators, bearings, friction materials, seals, springs
- Couplings - gears, seals, housing
- Slider Crank - bearings, rods/shafts, seals/gaskets, actuators
- Electric Motors - bearings, motor windings, brushes, armature shaft, housing, gears

The user should be familiar with the equipment and the Handbook so that the correct type and number of sub-components can be included in the model.

The prediction module also allows users to define maintenance tasks and assign these tasks to replaceable items in the prediction hierarchy. The program predicts MTTR (mean time to repair) values for sub-system and system levels in support of standards such as MIL-HDBK-472. Tasks are categorised into the following elements :

- Preparation
- Fault Isolation
- Disassembly
- Interchange
- Reassembly
- Alignment
- Checkout
- Start Up

The Prediction Module allows the user to interactively construct a block diagram which represents the structure of the system at various hierarchical levels. Blocks represent sub-systems or units and may contain constituent blocks, super-components or basic components. Super-components represent hybrids and interconnection assemblies. Basic components represent devices such as integrated circuits, relays and capacitors. The failure rates of components are determined by the Prediction Module by applying the appropriate standard model. The failure rates of blocks are determined by summing the failure rates of constituent blocks and components.

The Prediction Module also allows the user to define linked blocks. Linked blocks are blocks which have identical characteristics to other blocks in the hierarchical block diagram. This means that if, for example, you have a common PCB which is used in a number of different parts of the system, you only have to enter its total information once. There is also a powerful copy and paste facility which allows you to copy components or entire sub-systems to a different part of the block hierarchy and then modify any of the associated block or component parameters. The copy and paste facility may also be used to transfer data between different projects or from libraries.

Many of the component models use some ten or more parameters for the calculation of the component failure rate. Entering this information into The Prediction Module for each component and for each application of a particular component would be a very tedious task. The Prediction Module makes this task easy in many ways :

Default values for component parameters are set automatically for each component type. If you would prefer different default values you may change them.

Each component has a part number that you define (up to 30 characters long). Once you have specified any data for the component which differs from the default values you may place the component elsewhere in the system just by calling up the part number.

You may open an old project and call the component information into your new project by using the part number, or build up libraries using the library management facility. You may also use the off-the-shelf libraries available with the Reliability Workbench Program.

You may use the copy and paste facility provided to copy components or sub-systems between projects or within the same project.

The Prediction Module automatically calculates the failure rates associated with new components as they are added to the system. In addition the program automatically updates all dependent failure rates in the system as well as the overall system failure rate. At any time the user may also view the pi factors associated with a single component model.

The Prediction Module provides a powerful 'inheritance' system which allows you to make global changes to certain parameters such as environment or quality levels. Inherited parameters modified at block level will automatically be modified for connected blocks and components at lower levels with matching parameters.

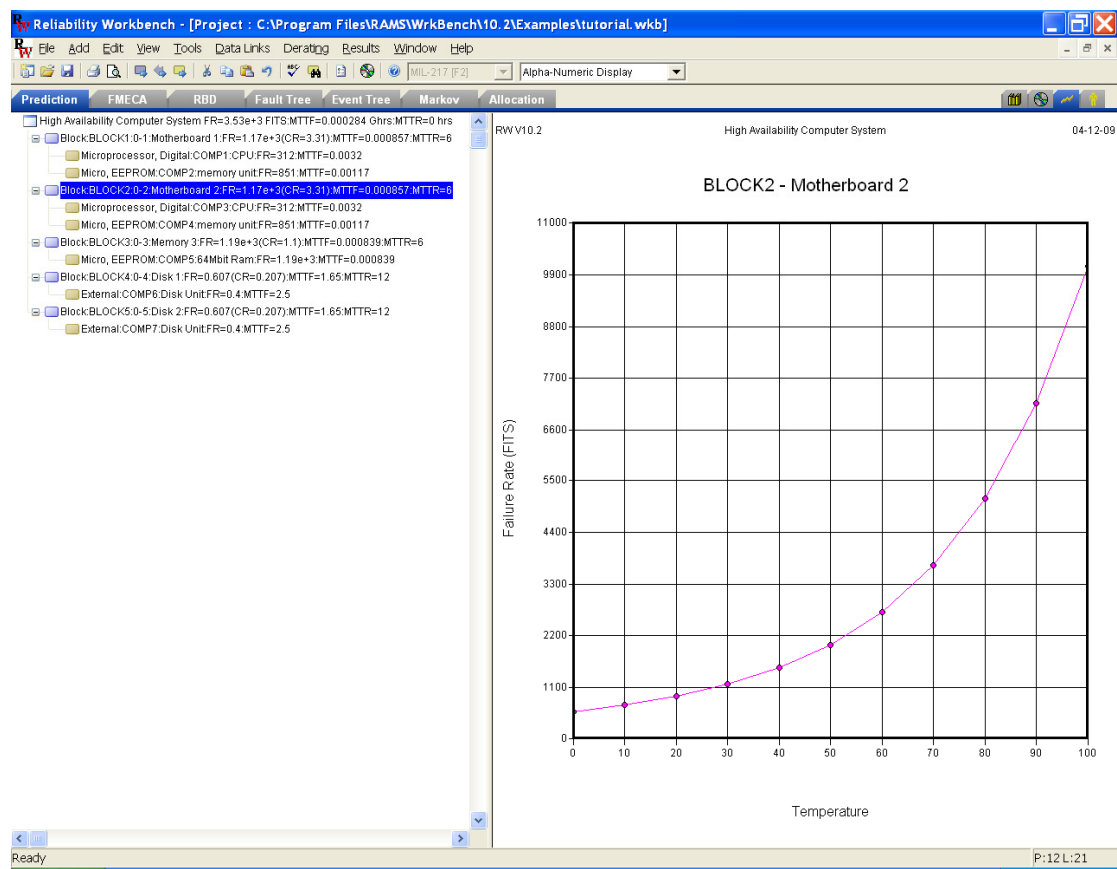
The Prediction Module also directly produces a number of graphs to show the effect of temperature, stress and environment changes on system, sub-system or unit failure rate predictions. Graphs may also be used to compare predicted failure rates during different operational phases or missions.

The Prediction Module also provides a powerful conversion facility which allows the user to automatically convert MIL-217 components into their Telcordia equivalent and vice versa.

Project data may also be automatically transferred to the FMECA, RBD or Fault Tree Modules. The Prediction Module block diagram hierarchy is transformed into the necessary failure mode, RBD or fault tree format automatically. Failure data is also transferred and a dynamic link optionally maintained.

Once you have entered your data you will naturally want to check descriptive text for spelling errors. The spelling checker facility will check individual text phrases or the entire project for errors and recommend suitable replacements. User-defined dictionaries may also be constructed to eliminate common technical words from being flagged. A global text replacement facility is also provided allowing you to replace an old text string with a new one throughout your Prediction project data.

A powerful Report Generator facility allows customised reports and graphs to be produced. Import and export facilities allow parts lists and other data to be transferred to and from databases, spreadsheets and files.



Prediction Module Screen Shot

FMECA Module Features

The FMECA Module of Reliability Workbench provides the full framework and reporting facilities to allow users to construct FMECAs to MIL-STD-1629A, BS 5760 Part 5 and similar standards as well as customising the FMECA to the user's own requirements. In addition Process and Design FMEAs and commercial aircraft FMEAs may also be constructed and analysed within this module. EFA format FMECAs may also be constructed.

FMECA

A Failure Mode, Effects and Criticality Analysis is a procedure for identifying potential failure modes in a system and classifying them according to their severity values. A FMECA is usually carried out progressively in two parts. The first part identifies failure modes and their effects (Failure Mode and Effects Analysis). The second part ranks failure modes according to the combination of severity and the probability of that failure mode occurring (Criticality Analysis).

The FMECA procedure may be summarised as completing the following steps :

- Define the system to be analysed
- Construct a hierarchical block diagram
- Identify failure modes at all levels of indenture
- Assign effects to the failure modes
- Assign severity categories to effects
- Enter other failure mode data such as failure detection methods, failure rates, etc.
- Rank failure modes in terms of severity and criticality
- Produce reports highlighting critical failures
- Recommend redesign or maintenance actions to reduce critical failures

Two alternative approaches may be used when performing a FMECA - a functional approach or a hardware approach. The functional approach considers sub-systems in terms of their function within the system and is often applied when hardware components cannot be uniquely identified. Blocks in the block diagram will represent sub-system functions at various levels of indenture when a functional approach is adopted. The hardware approach is usually adopted when hardware components can be uniquely identified in the system. Blocks in the block diagram will represent hardware components and sub-systems at various levels of indenture when a hardware approach is adopted.

The FMECA Module provides interactive graphical facilities for constructing a block diagram representing the logical connection between the sub-systems and components constituting the overall plant or system. This diagram may be extended to represent failure modes at various hierarchical levels. The FMECA Module permits the rapid entry of data using pop-up dialogs which may be accessed simply by clicking on the relevant part of the block diagram.

One of the most powerful features of the FMECA Module is its ability to automatically trace failure effects, severity values and failure causes through the system hierarchy. Failure rate and criticality values are automatically calculated by the program. The FMECA Module will also filter detectable and non-detectable failures in reports and determine the ratio between the frequency of detectable failures and total failures.

A large proportion of data entered when performing a FMECA is descriptive text. The FMECA Module provides a master phrase library which contains commonly used descriptions of component parts, failure modes and effects. These phrases may be inserted into descriptive fields by selecting the required phrase from the library saving considerable typing and ensuring consistency. Users may build up their own phrase libraries or add to the master library provided.

The FMECA Module provides an apportionment library facility which allows the user to create commonly used component and failure mode groupings. Each failure mode in a grouping is given an apportionment percentage. Users may add a component to the FMECA block diagram together with the appropriate failure modes by selecting the appropriate entry in the apportionment library. This saves

considerable effort when constructing the block diagram. The FMECA Module is provided with an apportionment library derived from MIL-HDBK-338.

Process and Design FMEA

Process and Design FMEAs provide an alternative approach to performing a Failure Mode and Effect Analysis. Occurrence, severity and detection rankings replace apportionments and failure rates. Risk Priority Numbers (RPN) are calculated by multiplying the severity, occurrence and detection ranking numbers together.

Commercial Aircraft FMEA

Commercial Aircraft FMEAs are a special format of FMEA allowing users to define failure modes and effects at unit, system, engine and aircraft level.

General Features

The FMECA Module provides a wide range of functions for speeding-up the construction of a FMECA project. Data may be easily transferred within the same project or between different projects using the cut, copy and paste facilities. Search and filter facilities allow data to be easily located. A wide range of view options allows different data types to be displayed in the block diagram. Blocks may be opened and closed in the block diagram to reveal or hide connected data.

The FMECA Module also provides a project link facility that allows a group of users to work independently on sub-projects which are later linked into a master project. In addition users may append data from one project to another in a few seconds.

Once you have entered your data you will naturally want to check descriptive text for spelling errors. The spelling checker facility will check individual text phrases or the entire project for errors and recommend suitable replacements. User-defined dictionaries may also be constructed to eliminate common technical words from being flagged. A global text replacement facility is also provided allowing you to replace an old text string with a new one throughout your FMECA project data.

A powerful Report Generator facility allows customised reports and graphs to be produced. Import and export facilities allow data to be transferred to and from databases, spreadsheets and files.

Some of the FMECA Module features are summarised below :

- Apportionment library for fast data entry
- Phrase library for eliminating repetitive text entry
- User-definable data fields
- Effects automatically traced through to system level
- Root contributors automatically calculated
- Extensive sorting facilities
- Data verification for consistency checks
- Severities automatically assigned
- Automatic criticality calculations
- Multiple failure effects permitted for a single failure mode
- Effects may be defined on any higher-level sub-system
- User-definable severity category libraries
- Failure detection filter facility
- Sub-projects may be linked to a master project
- Automatic data transfer from prediction modules
- Drag and drop components to the RBD module
- Spelling checker and global text replacement
- Automatic fault tree construction facility
- Automatic conversion to RCMCost project
- Calculation of Safe Failure Frequency using IEC 61508-6 diagnostic coverage models

RBD Module Features

The structure of a reliability block diagram (RBD) defines the logical interaction of failures within a system. Individual blocks may represent single component failures, sub-system failures and other events that may contribute towards system failures. The reliability behaviour of an individual sub-system block may be represented by a RBD at a lower hierarchical level.

The logical flow of a RBD originates from an input node at the left hand side of the diagram to an output node at the right hand side of the diagram. Blocks are arranged in series and parallel arrangements between the system input and output nodes.

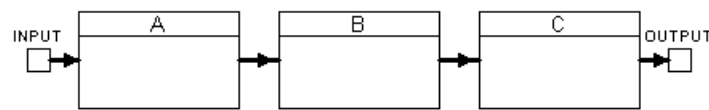
For the system to be successful in its operation at least one path must be maintained between the system input and output nodes. A simple series arrangement of 3 blocks A, B and C would only require one of the blocks to fail to eliminate the single success path from input to output node. A simple parallel arrangement of 3 blocks A, B and C would require all 3 blocks to fail to eliminate the 3 success paths from input to output node. Expressed in Boolean Algebra nomenclature (where the '+' symbol represents OR logic and the '.' symbol represents AND logic) we have :

For A, B and C in series :

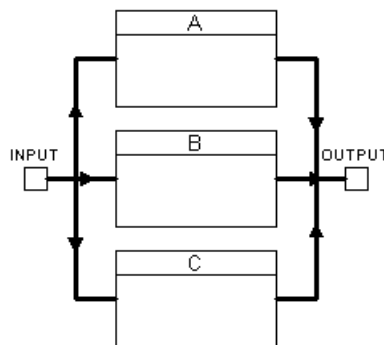
$$SystemFailure = A + B + C$$

For A, B and C in parallel :

$$SystemFailure = A.B.C$$

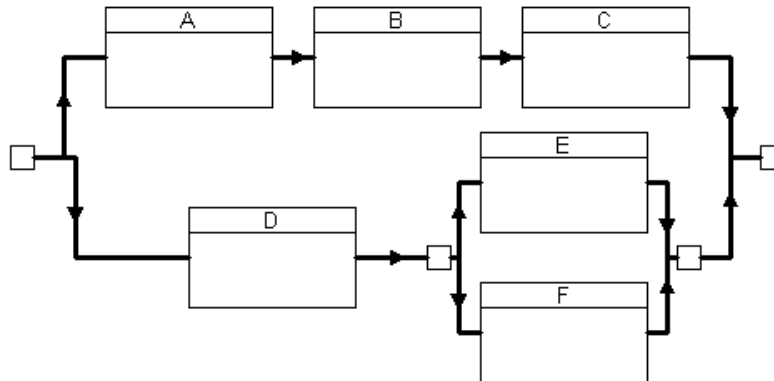


Simple Series Arrangement



Simple Parallel Arrangement

In practice RBDs will consist of combinations of series and parallel arrangements.



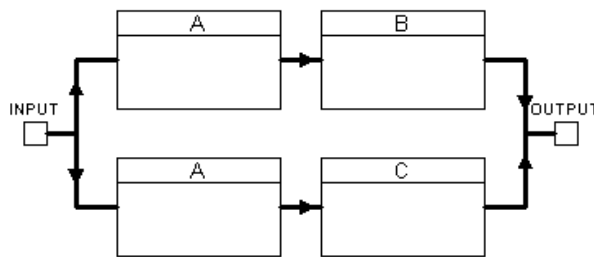
Block Diagram with Combined Series and Parallel Arrangements

For the combined series and parallel arrangement illustrated above :

$$SystemFailure = A.D + A.E.F + B.D + B.E.F + C.D + C.E.F$$

RBDs can represent the effect of common cause failures by repeating the same block in different parts of the diagram. Consider a simple parallel-serial configuration with block A repeated in both serial lines. Failure of block A would eliminate both success paths at the same time resulting in the following Boolean Algebra expression for system failure.

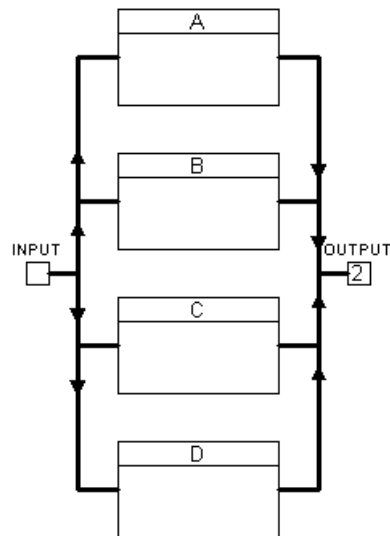
$$SystemFailure = A + B.C$$



Simple Block Diagram with Common Failure A

RBDs may also be used to represent voting arrangements. Nodes to the right of a parallel arrangement may be given a vote number to indicate how many success paths must be available through the parallel arrangement (if a vote number is not specified only one path need be available). A simple parallel arrangement of 4 blocks A, B, C and D with a vote number (number of available paths required for success) of 2 would result in the following Boolean Algebra expression for system failure.

$$SystemFailure = A.B.C + A.B.D + A.C.D + B.C.D$$



Simple 2-out-of-4 vote arrangement

The Boolean Algebra expressions used above are all minimal in that each term represents a minimum combination of failures required to cause a system failure.

For example in the expression

$$SystemFailure = A + B.C$$

there are two terms :

$$A \text{ and } B.C$$

The first term indicates that block A failing will result in system failure. If block A and block B were to fail together this would also result in the system failing. However, the failure of block B is redundant information as the failure of the system is not dependent on block B failing once block A has already failed. Each minimal term in the Boolean Algebra expression is referred to as a minimal cut set and represents a minimal combination of failures which will cause the system to fail.

The RBD Module is a powerful systems reliability analysis tool that allows reliability block diagram analyses to be performed in an integrated environment.

The RBD Module is capable of analysing large and complex RBDs producing the full minimal cut set representation for identified systems and sub-systems.

The RBD Module calculates a range of importance measures as well as providing standard system and sub-system parameters such as unavailability, unreliability, number of expected failures etc. The program allows users to construct a single project database containing failure model data and block diagrams representing one or more systems. Large block diagrams may be split into sub-systems (there is no limit to the number of hierarchical levels that may be specified for a project). Navigation between sub-systems is easily achieved using the Change Page or Find facilities provided by the program.

The RBD Module uses efficient minimal cut set generation algorithms to analyse large and complex RBDs. Markov analysis capabilities are also provided for analysing standby groups of blocks. Hot, cold and warm standby systems may be modelled taking into account the effect of maintenance queuing.

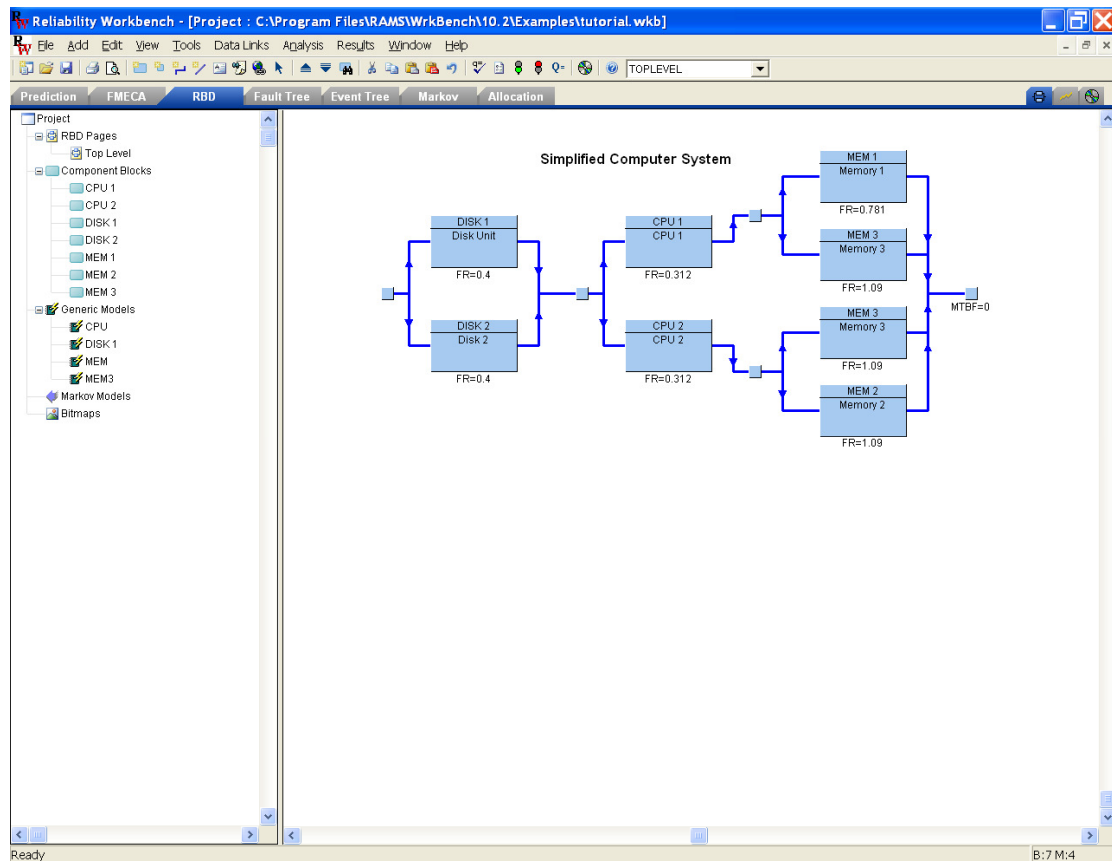
The RBD Module also includes a special Beta Factor Common Cause Failure (CCF) facility that allows users to associate groups of blocks with the same CCF model. During the analysis special CCF events are automatically generated by the program allowing the accurate evaluation of the effects of common cause failures.

The RBD Module automatically evaluates the system minimal cut sets and uses the cut sets to determine system performance parameters.

The RBD module now allows the user to select the IEC-61508-6 method of handling dormant failures. This provides an alternative method of calculating the unavailability of cut sets containing more than 1 dormant failure.

The program produces high quality reliability block diagram reports. Automatic pagination allows the user to quickly construct and print large RBDs.

A powerful Report Generator facility allows customised reports and graphs to be produced. Import and export facilities allow data to be transferred to and from databases, spreadsheets and files.



Typical RBD Module Screen Display

Fault and Event Tree Module Features

The fault and event tree modules of Reliability Workbench provide a fully interactive graphics and analysis facility for performing integrated fault tree and event tree analyses. The modules are capable of analysing large and complex fault and event trees producing the full minimal cut representation for fault tree TOP events and event tree consequences.

The modules provide CCF analysis, importance analysis, uncertainty and sensitivity analysis facilities. The program allows users to construct a single project database containing generic data and event tables, fault trees with multiple TOP events, event trees originating from different initiating events, CCF tables and consequence tables. Fault and event tree pagination is automatically controlled by the program. Fault tree TOP events may be used to represent specific nodes in the event tree. Multiple branches are also handled to allow for partial failures.

Reliability Workbench uses efficient minimal cut set generation algorithms to analyse large and complex fault and event trees. NOT logic may be included in the fault and event trees at any level and event success states retained in the analysis results as an option.

Fault and Event Tree Construction

Reliability Workbench provides a fully interactive environment for constructing and editing fault and event trees.

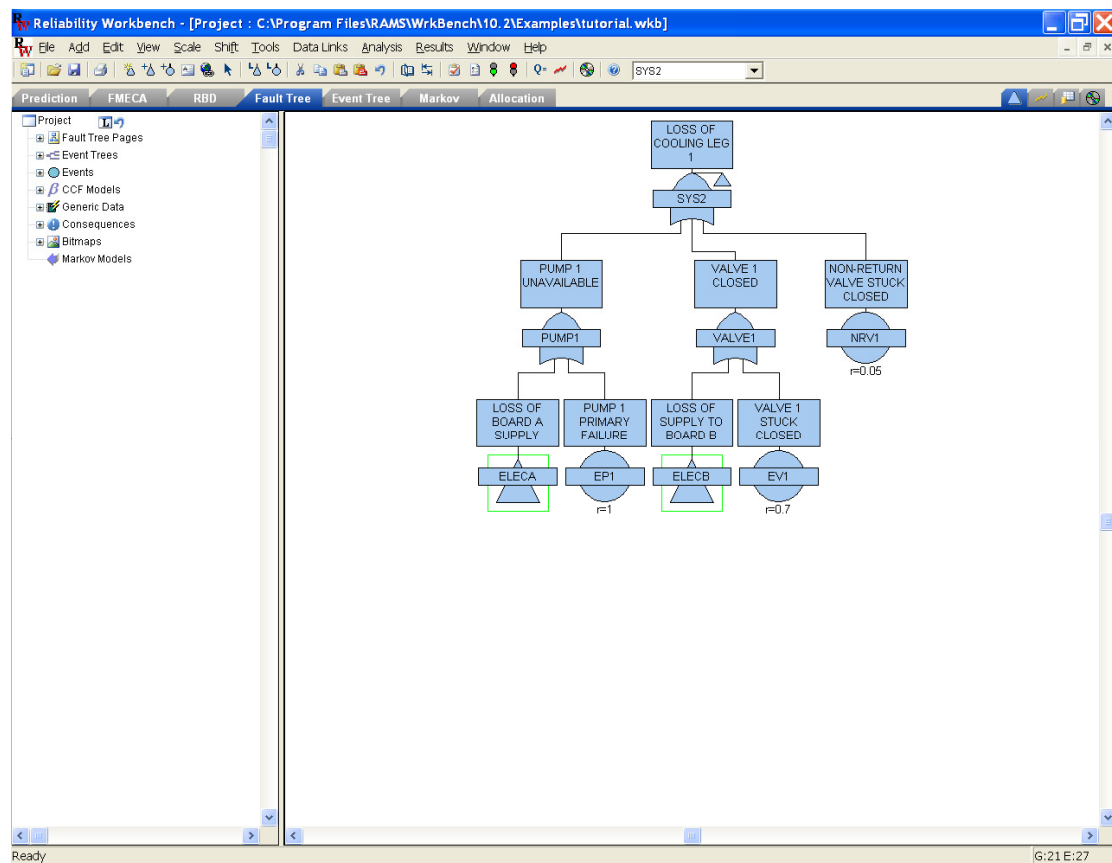
Features include:-

- Automatic drawing facilities produce high quality diagrams without any effort from the user
- Time saving features such as double mouse clicks to bring up gate, event and branch attributes
- Drag and drop add mode for fast tree construction
- Extensive diagram scale and shift options including manual shifting of sub-trees and automatic alignment to the screen edit area
- Global and local font selection allowing highlighting of labels and descriptions
- Automatic paging facilities - simply identify gates or branches with a new page tag and the program takes care of pagination
- Append facilities for fault trees produced by different users
- OR, AND, VOTE, NOT, Exclusive Or, Inhibit and Priority AND gates supported
- Basic, Conditional, Undeveloped, Dormant and House basic event symbols supported
- Multiple branching supported for event trees
- Extensive on-line help facility including key word search
- Attributes such as event parameters, generic model codes, branch names and column probabilities may be displayed on diagrams if required
- Cut, copy and paste facilities available for fault tree symbols
- Flexible labelling formatting allows the user to place descriptive text anywhere within a fault or event tree page
- Project database tables may be easily edited using direct and dependency filtering
- Event and gate names may be globally edited
- Circular logic checks during fault tree construction
- Import and export facilities for data models
- Undo and automatic backup facilities
- Delete hidden data facility for tidying-up large projects

Analysis

- Range of event failure and repair models including fixed rates, dormant, sequential, initiator, standby, binomial and Poisson failure models
- Fault tree house event analysis
- Full minimal cut set analysis (including success states if required)
- CCF analysis using the beta factor, MGL, alpha factor or beta BFR methods

- Post-processing facilities for accurate upper bound calculations
- Importance analysis with Fussell-Vesely, Birnbaum, Barlow-Proschan and Sequential importance measures. Weighted values provided for event tree consequences
- Initiator-enabler analysis for sequence dependent analyses
- Uncertainty analyses allowing confidence levels to be determined from event failure and repair data uncertainties
- Sensitivity analysis allowing the automatic variation of event failure and repair data between specified limits
- Time dependent analysis providing intermediate values for time dependent system parameters
- Verification checks providing diagnostic information before commencing an analysis. Checks are made for circular logic, undefined gates, invalid initiators etc.
- Status facility to indicate whether analysis results are out of date with respect to project data
- combinatorial cut set analysis for cases where basic event probabilities are relatively small compared to the cut-off values set by the user.
- Option to use the IEC-61508-6 method of handling dormant failures to calculate the unavailability of cut sets containing more than 1 dormant failure.



Event Tree Module Screen Shot

Markov Module Features

Then Markov module of Reliability Workbench analyses state transition diagrams using numerical integration techniques. The module provides facilities for defining multiple phases representing continuous or discrete transitions. The program also analyses non-homogeneous processes by allowing time-dependent transition rates to be defined. Systems with time-dependent transition rates are strictly non-Markovian, however the addition of this facility in the program allows certain types of ageing processes to be modelled.

The Markov module uses 4th order Runge-Kutta numerical integration techniques to analyse the Markov diagram. The system logic is represented by a state transition diagram that may be easily constructed using the program's interactive graphics facilities. The system lifetime may be split into phases with different transition rates.

Facilities provided by the module are summarised below :

- Graphically constructed transition diagram
- Division of analysis into separate phases
- State attribute editing via easy-to-use dialogs
- Data verification for consistency checks
- Time-dependent transition rates modelled
- Global parameter facility for repetitive data
- Calculation of a wide range of probabilities and frequencies
- Comprehensive reports interfacing with Microsoft Word, Excel etc.
- Graphs and plots showing time-dependent results

Markov Analysis Methods

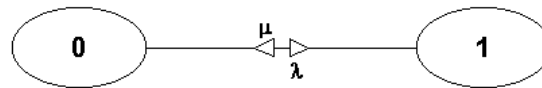
Markov analysis provides a means of analysing the reliability and availability of systems whose components exhibit strong dependencies. Other systems analysis methods (such as the Kinetic Tree Theory method employed in fault tree analyses) generally assume component independence which may lead to optimistic predictions for the system availability and reliability parameters. Some typical dependencies that can be handled using Markov models are

- Components in cold or warm standby
- Common maintenance personnel
- Common spares with a limited on-site stock

The major drawback of Markov methods is that Markov diagrams for large systems are generally exceedingly large and complicated and difficult to construct. However, Markov models may be used to analyse smaller systems with strong dependencies requiring accurate evaluation. Other analysis techniques, such as fault tree analysis, may be used to evaluate large systems using simpler probabilistic calculation techniques. Large systems that exhibit strong component dependencies in isolated and critical parts of the system may be analysed using a combination of Markov analysis and simpler quantitative models.

The state transition diagram identifies all the discrete states of the system and the possible transitions between those states. In a Markov process the transition frequencies between states depends only on the current state probabilities and the constant transition rates between states. In this way the Markov model does not need to know about the history of how the state probabilities have evolved in time in order to calculate future state probabilities. Although a true Markovian process would only consider constant transition rates the Markov module does allow time-varying transition rates to be defined. These time-varying rates must be defined with respect to absolute time or phase time (the time elapsed since the beginning of the current phase).

In order to illustrate the use of Markov methods let us consider a very simple Markov model. The Markov diagram below represents the failure and repair behaviour of a single component.



The component has two states only, the working state (State 0) and the failed state (State 1). It is a repairable component (with failures immediately revealed) and therefore the component may move from the failed state to the working state as well as moving from the working to failed state. These possible transitions are represented by the transition lines and arrows in the Markov diagram.

The Markov diagram represents the logical behaviour of a component or system and should contain all possible states and transitions for the component or system under given conditions.

The Markov diagram above may be translated into a set of linear differential equations that represent the time-dependent behaviour of the state probabilities. These equations are given below.

$$\frac{dP_0(t)}{dt} = -\lambda P_0(t) + \mu P_1(t)$$

$$\frac{dP_1(t)}{dt} = \lambda P_0(t) - \mu P_1(t)$$

where $P_i(t)$ = probability of being in state i at time t
 λ = component failure rate
 μ = component repair rate

Integration of these equations after applying the initial conditions

$$P_0(0) = 1$$

$$P_1(0) = 0$$

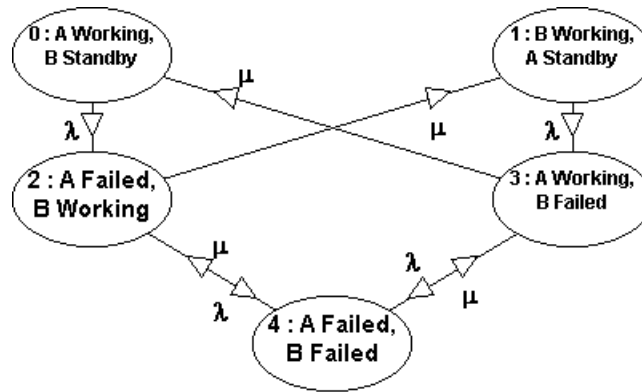
produces the well-known expression for the unavailability of a two-state repairable component with immediately revealed failures :

$$P_1(t) = \frac{\lambda}{\lambda + \mu} (1 - e^{-(\lambda + \mu)t})$$

As t becomes very large the component unavailability approaches the steady state solution of

$$P_1(\infty) = \frac{\lambda}{\lambda + \mu}$$

The Markov diagram below represents the failure and repair behaviour of a 2-pump standby system. The diagram assumes that the pumps are identical and that there is no possibility of a pump failing if it is in standby (cold standby).



Only one pump is required to be working at any time to provide full functionality. If the operating pump should fail, the standby pump will be started and the failed pump will be repaired. There is therefore a dependence between the two pumps.

Even for this small system of two components it can be seen that the number of states in the Markov model is rapidly increasing. The steady-state solution for the unavailability of the two-component system is equal to the steady-state probability for state 4 :

$$P_4 = \frac{\lambda^2}{\lambda^2 + 2\lambda\mu + 2\mu^2}$$

As the size of the Markov diagram increases the task of evaluating the expressions for time-dependent unavailability by hand becomes impractical. Computerised numerical methods may be employed, however, to provide a fast solution to large and complicated Markov systems. In addition these numerical methods may be extended to allow the modelling of phased behaviour and time-dependent transition rates. MKV employs a Runge-Kutta 4th order numerical integration technique to determine the time-dependent behaviour of state probabilities. The time step employed during the integration may be specified by the user. MKV also provides three different error indicators to allow the user to assess the accuracy of the result.

Continuous Time and Discrete Transition Phases

MKV allows the user to split the system lifetime into discrete fixed-interval phases. Each phase may be represented by a set of transitions unique to that particular phase. States may not vary between phases. Phases may be specified as *continuous time* phases or *discrete transition* phases. Continuous time phases have transitions which are quantified with transition rates. Transition rates are generally failure and repair rates. Continuous time phases have finite phase durations. Discrete phases do not have a phase duration associated with them as they represent fixed probability transitions between states. They may be used to represent fixed interval inspections and preventive maintenance actions. The transitions in a discrete phase must be identified with fixed probabilities.

For continuous time phases the user may specify transition rates which vary with absolute system time or absolute phase time. The time-varying transition rates are specified in the form of a Weibull distribution which is superimposed on the base failure rate :

$$\lambda(t) = \lambda_0 + \frac{\beta(t - \gamma)^{\beta-1}}{\eta^\beta}$$

- where λ_0 = base failure rate
- η = Weibull characteristic lifetime
- β = Weibull shape parameter
- γ = Weibull location parameter

Calculated Parameters

MKV calculates a wide range of system parameters during the integration process. These parameters are

- Unavailability
- Availability
- Unreliability
- Reliability
- Failure frequency (unconditional failure intensity)
- Repair frequency (unconditional repair intensity)
- Conditional failure intensity
- Conditional repair intensity
- Number of expected failures
- Number of expected repairs
- Mean unavailability over lifetime
- Mean availability over lifetime
- Expected total downtime over lifetime
- Expected total uptime over lifetime

MKV also calculates mean and lifetime probabilities for states in the transition diagram.

New Features for Version 10.3

Certificate Licensing

Reliability Workbench v10.3 is licensed using the Flexnet v11.7 certificate licensing systems. This system uses an activation ID to activate the software license over the internet (a file activation method is available for machines that do not have an internet connection). This upgrade corrects two known issues with the previous version of Flexnet:

- Composite ID of machine changes if there is a change to machine hardware (e.g. network adapter is connected/disconnected), thus invalidating the license
- On rare occasion the borrowing facility in Flexnet v10 fails to work