# A Deep Dive into System Modeling Using Reliability Block Diagram (RBD) Analysis

Reliability Block Diagram (RBD) analysis is a methodology for assessing and calculating the reliability and availability metrics of complex systems using a graphical depiction of the system. RBD techniques are most often employed to model complex systems, especially those that incorporate redundant components and repair data. Within RBD, there are several component models and calculation metrics available to ensure complete and accurate system analysis is obtained. In this whitepaper, we will discuss these various options in technical detail.

RELYENCE CORPORATION

# Table of Contents

# System Modeling with RBD Analysis

In our blog post *A Guide to System Modeling Using Reliability Block Diagram (RBD) Analysis*, we explored the fundamentals of RBD analysis and how it can be used to calculate a variety of system reliability and availability metrics. We discussed the benefits of RBD, how it can be used in certain fields, and detailed the steps involved in performing RBD analysis. For further information on the basics of RBD analysis, be sure to check out the blog post or the Relyence RBD product webpages. This whitepaper will explore in detail the system modeling procedures and options that were introduced previously.

# Component Modeling & Organization

There are several important elements to consider when performing RBD analysis. We will take a look at some of the most useful features available to make your analyses most effective. They include redundant configurations, switching characteristics, failure and repair distributions, and subdiagram organization.

## Redundant Configurations

One of the main benefits of RBD analysis is that it allows you to perform reliability and availability calculations for systems that include redundant components. Redundancy in terms of RBD analysis means that when a component or path fails, a secondary component or path can take over to keep the system up and running. This is a key difference compared to Reliability Prediction analysis where a component or system's failure characteristics are calculated independent of any redundancies.

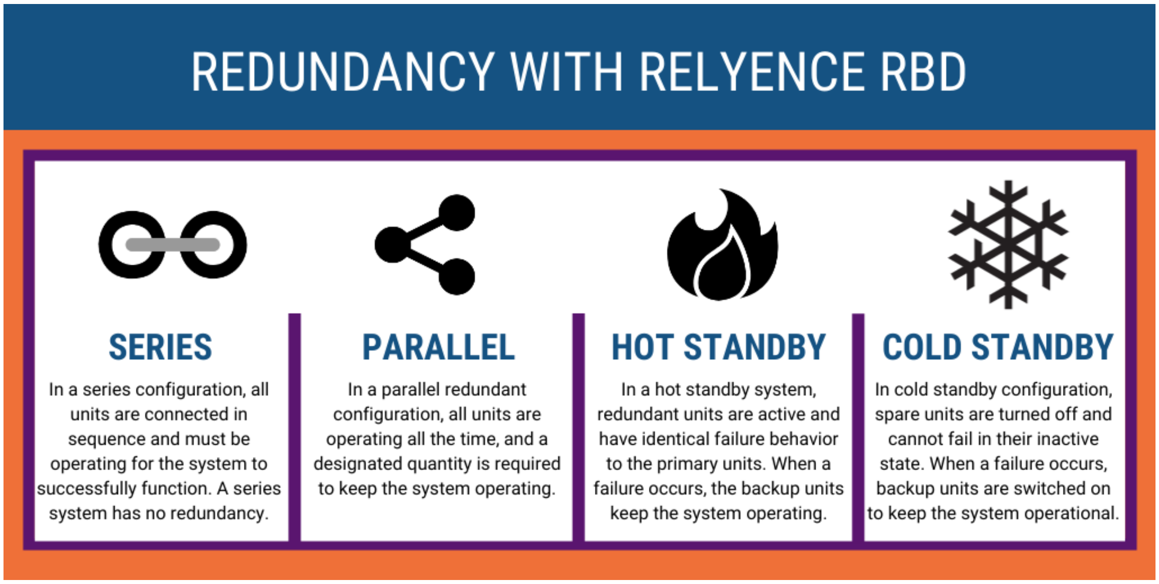
When redundant relationships need to be modeled in RBD, redundancy type and switching behavior are taken into account. Redundancy type is specified by the component configurations that are defined in the graphical diagram. Classifications and explanations of the redundancy types available in RBD include:

- Series
    - A series configuration is the simplest form for an RBD. A system in series has no redundancy and all components connected in series must be operating in order for the system to successfully function.
    - Some examples of series systems can include a smartphone, a PC, and an RC helicopter.
- Parallel
    - In a parallel, also referred to as k-out-of-n, configuration, all components are operating at all times but only the designated quantity (k) out of the total quantity of components (n) is required to keep the system fully operational. This is a basic form of redundancy where there are additional units on hand to take over should a given number of components fail. An example of parallel redundancy is when two components are included in your system in a side-by-side configuration and both are operating. Since only one is required for successful system operation, the second component is one hand to take over if the first component fails.
    - A real-world example of a parallel system is a commercial twin engine aircraft that is built to continue operating in the event of a single engine failure.
- Standby
    - In a standby configuration, components are designated as either active or spare units. Similar to a parallel configuration, the spare components are on hand to take over should any of the active units fail. There can exist a similar k-out-of-n relationship in standby systems where only a designated quantity (k) out of total quantity (n) are required for the system to operate. In addition, standby systems allow switching failures and delays to be considered. There are different classifications of standby configurations based on the failure characteristics of the spare units:

- Hot Standby: Spare units are turned on and have identical failure and repair characteristics to the active units. Because of this, Hot Standby systems function identically to parallel systems if there is no switching failure or delays to take into account.

- Cold Standby: Spare units are turned off and cannot fail while they are in their inactive (or quiescent) state. Once the failure of an active unit occurs, the appropriate number of spares are switched on and, at that point, have non-zero failure rates. Cold Standby spare units are assumed to have the same failure rate as active units once in operation.

- Standby systems include Uninterruptible Power Supplies (UPS) and backup generators.



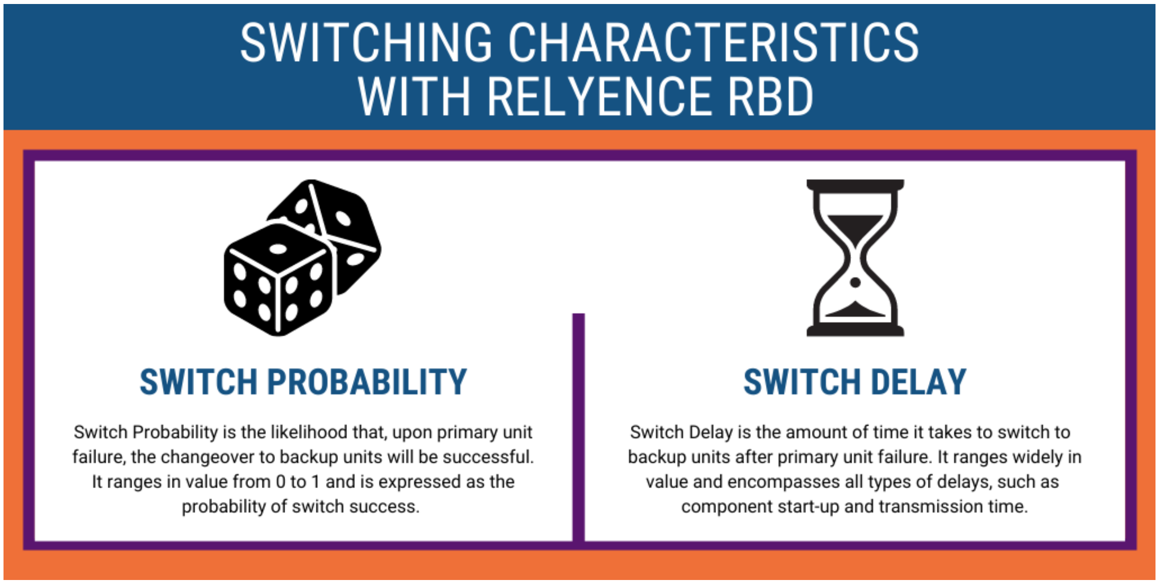

## Switching Characteristics

In systems involving redundant components, switches can be implemented to change to backup components upon primary unit failure. Ideally, switches would always successfully connect to the backup units and make the change over immediately. However, depending on the switching mechanism there may be some chance of switch failure or switch delay. For the most accurate component modeling, these variables must be taken into account in RBD analysis if appropriate.

- Switch Probability: The probability that, upon primary unit failure, the switch to backup units will be made successfully. Switch Probability ranges from 0 to 1 and is the percentage chance of switch success – i.e., a value of 0.5 means that the switch is successful 50% of the time. With a value of 1, the switch to the backups will always be successful. For values less than 1, there can be failures when trying to connect to the backup units. Depending on the system configuration, these switching failures can lead to overall system failures.

- Switch Delay: The amount of time the switchover to backup components takes to occur. With a value of 0, the switch happens instantaneously after primary unit failure. Switch delay values can vary widely depending on the type of system analyzed and should include all potential types of delays, such as startup, transmission, and logistic, if applicable. Depending on the system configuration, switch delay values greater than 0 can lead to overall system failure and increased downtime while the switchover is in progress.

Switch Probability is applicable when modeling parallel and standby redundancy and, additionally, Switch Delay is included when modeling standby components.



## Failure and Repair Distributions

When modeling systems using RBD techniques, common statistical profiles are used to describe the failure and repair behavior of individual components. Fitting failure and repair times to statistical distributions can be done in several ways:

- If using predictive tools, such as Reliability Prediction or Maintainability Prediction analysis, the calculated Failure Rate, MTBF, and MTTR can be used in RBD.
- If using statistical fitting tools, such as Weibull or ALT analysis, the calculated distribution coefficients can be used to model failure and repair characteristics in RBD.
- If using a corrective action tracking system, such as FRACAS, field-based MTBF and MTTR data can be used in RBD.
- If not using other reliability tools, other estimation methods such as known similar product performance, maintenance expertise, or manufacturer specifications can be used in RBD.

By supporting a range of distributions, RBD allows you to most accurately model the characteristics of your components for precise analysis. Commonly used distributions, their definitions, and associated parameters include:

- Constant Time: Failures occur at constant time intervals through a component's operating life. For example, failures occur after every 100 hours of operation. Repairs take a constant amount of time to complete. For example, repair time is 10 hours.
- Exponential: Failure Rate is constant for component's operating life (i.e., the failure distribution is based on Failure Rate or MTBF). Repair Rate is constant when a component is undergoing its repair process (i.e., the repair distribution is based on MTTR).
- Normal: Also referred to as Gaussian distribution, it is a symmetric distribution centered at a component's mean time to failure or mean time to repair (μ) with a standard deviation (σ) that defines the spread of the distribution curve.

- Lognormal: Shares some similarities to the Normal distribution and is also based on a component's mean time to failure or repair (μ) and standard deviation (σ). In a lognormal distribution, the logarithm of the failure and repair values are normally distributed.

- Gumbel+: A type of Extreme Value Distribution (EVD) based on the Location parameter (ζ) and Scale parameter (δ). Gumbel+ (maximum Gumbel distribution) is based on the largest extreme value in a data set.

- Gumbel-: A type of Extreme Value Distribution (EVD) based on the Location parameter (ζ) and Scale parameter (δ). Gumbel- (minimum Gumbel distribution) is based on the smallest extreme value in a data set.

- Rayleigh: A type of statistical distribution based on Characteristic Life parameter (η) and Location parameter (γ). The Rayleigh distribution is equivalent to a Weibull distribution with the Slope parameter (β) equal to 2.

- Time Independent: Failure characteristics are based on a component's Unreliability. (Note that Time Independent is not available as a repair distribution.)

- Uniform: Lower and Upper Bounds define the outer limits of the failure and repair times. Inside of those bounds, time values are assumed to be distributed uniformly.

- Weibull: A type of statistical distribution commonly used in reliability engineering that is based on Slope parameter (β), the Characteristic Life parameter (η), and Location parameter (γ).

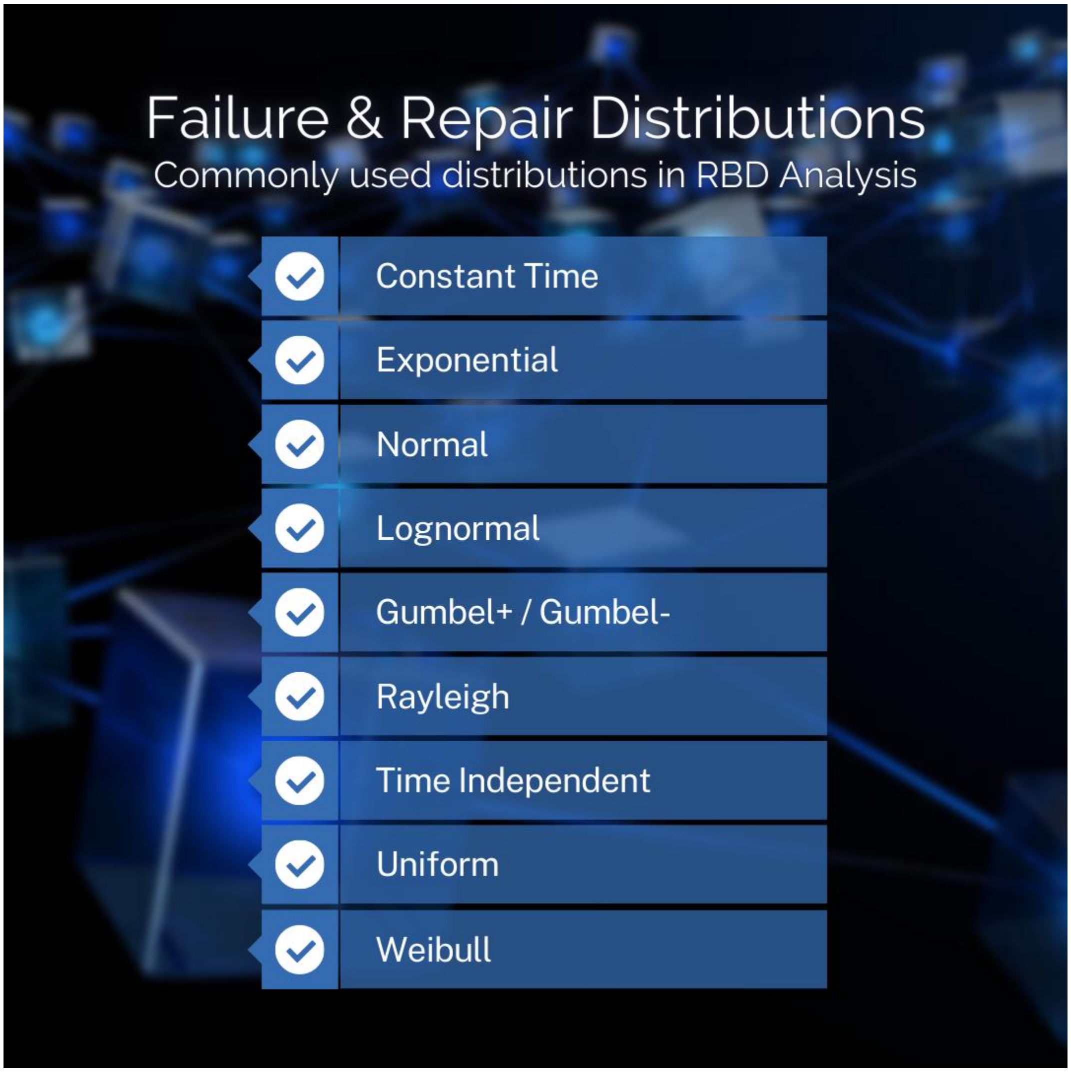
    - β defines the Slope parameter. Failure and Repair rate trends are based on its value:

        - $\beta < 1$ – Failure or Repair rate decreases over time

        - $\beta = 1$ – Constant Failure or Repair rate (Exponential distribution)

        - $\beta > 1$ – Failure or Repair rate increases over time

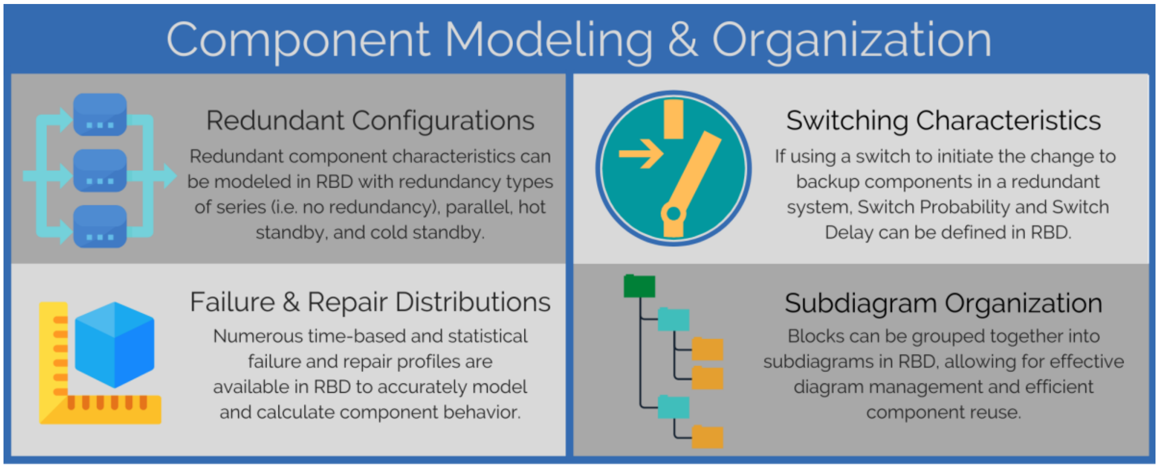
- η defines the Characteristic Life parameter. Depending on its value, eta will either stretch (larger η values) or contract (smaller η values) the width of the distribution curve.
- γ defines the Location parameter. This is time at which the first failures occur or the minimum time a repair takes.



### Subdiagram Organization

analysis allows components to be grouped into subdiagrams for organized diagram management. A subdiagram is most often a group of components that comprise an individual subsystem under the main system. For example, if you are analyzing a car, you may set up subdiagrams to model the engine, steering assembly, braking system, etc. Subdiagrams can be given their own RBD and are represented in their parent diagram by a single block. In this way, subdiagrams help to manage, organize, and maintain your RBD by offering a more compact visual view of your overall system. Subdiagrams can be especially useful to more efficiently represent complex diagrams.

An additional benefit to subdiagrams is to support reusability in cases where a subsystem is used in more than one place. Subdiagrams allow you to build one subsystem RBD that will be shared in all places that subdiagram is referenced.




## RBD Calculation Results

To perform its calculations, RBD analysis relies on a powerful calculation engine that takes into account the component arrangement and all defined failure and repair models. Depending upon the complexity of the diagram, RBDs may be evaluated using analytical techniques, Monte Carlo simulation, or a combination of the two. RBD calculation engines support a host of both time-based and steady state reliability and availability related metrics. Calculated metrics can be selected depending on the specific needs or requirements of the system under consideration.

In addition to system reliability and availability metrics, RBD analysis can include computations that help identify key components or groups of components in a given system. Results from these calculations can be used to denote paths of successful system operation or to discover critical components or groups of components that can lead to system failure.

## Time-Based Metrics

- Reliability: The probability that the system is functioning at a given time point. Calculating reliability does ***not*** take any potential repair processes into account.

- Unreliability: The probability that the system has failed at a given time point. Calculating unreliability does ***not*** take any potential repair processes repairs into account.

- Failure rate: Given that the system is currently operating, Failure Rate (λ) is the rate at which the system fails and is expressed in units of Number of Failures per Unit Time. Calculating Failure Rate does ***not*** take any potential repair processes into account.

- Equivalent failure rate: The failure rate a constant failure rate system would have to achieve to meet the same Reliability value at the same time as the system under analysis. Equivalent failure rate is calculated by solving the below equation for λ given a specified Reliability and Time:

$$R(t) = e^{-\lambda t}$$

- Availability: The probability that the system is operating at a given time point. Repairable component characteristics ***are*** taken into account when calculating Availability.

- Unavailability: The probability that the system is not operational at a given time. Repairable component characteristics ***are*** taken into account when calculating Unavailability.

- Mean availability: The average availability the system has experienced at that point in time.

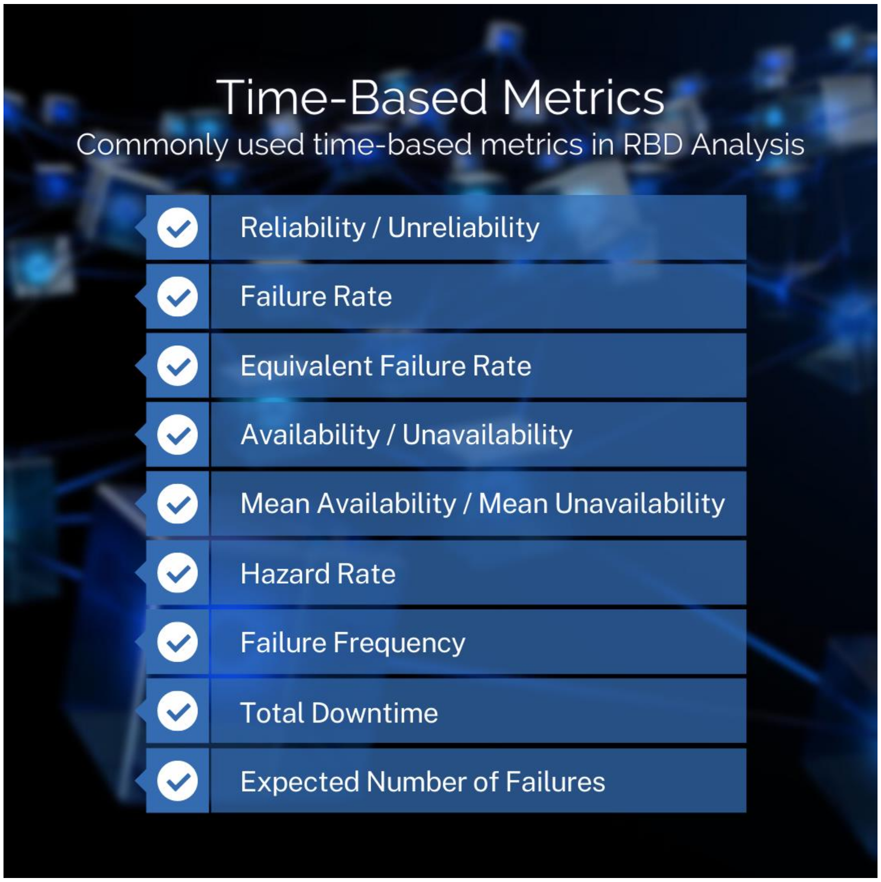
- Mean unavailability: The average unavailability the system has experienced at that point in time.

- Hazard rate: Given that the system is currently operating, Hazard Rate is the rate that the system fails and is expressed in Number of Failures per Unit Time. Repairable component characteristics ***are*** taken into account when calculating Hazard Rate.

- Failure frequency: Failure Frequency is the expected frequency at which failures will occur and is expressed in units of Number of Failures per Unit Time. Repairable component characteristics and system operating status ***are*** taken into account when calculating Failure Frequency.

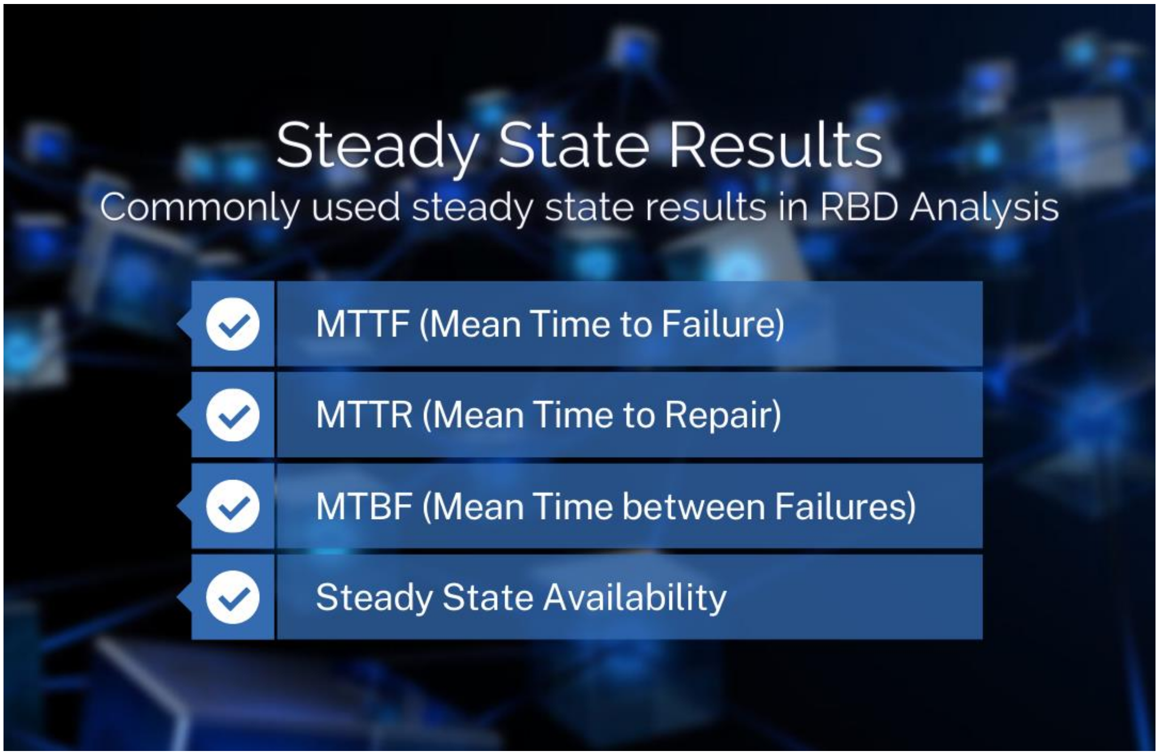
- Total downtime: The cumulative total of the time the system has been in a failed state.

- Expected number of failures: The cumulative total amount of times the system has failed up to a given time point.



## Steady State Results

- MTTF (Mean Time to Failure): The average amount of time to reach system failure. Calculating MTTF does ***not*** take any potential repair processes into account.
- MTTR (Mean Time to Repair): The average amount of time it takes to repair the system after it has reached a failed state.
- MTBF (Mean Time Between Failures): The average amount of time between consecutive system failures. MTBF includes both the time to fail and the time to repair.
- Steady State Availability: The availability the system will stabilize at if it operates sufficiently long enough (i.e., reaches steady state).



## Calculation Inputs

In addition to the above metrics, additional inputs that are often considered when performing RBD calculations include:

- End time and Number of display steps: RBD calculations all begin at time zero and will be computed out to the time you select as the End time. The interval between time point results is determined based on the Number of display steps selected. For example, when setting End Time = 1000 and Number of Display Steps = 11, the time point results will show 10 results plus time point 0. That is, results will be shown at time points of 0, 100, 200, 300, 400, 500, 600, 700, 800, 900, and 1000 hours.

- Number of iterations: The number of iterations to run if simulation techniques are required. Increasing the number of simulations increases the accuracy of the results but also increases computational time.

- Number of failures to reach steady state: Sets the number of failures components can experience before considered to have reached steady state.

- Set random number seed: Specifies the random number seed that is used in the simulation engine. Setting a random number seed allows you to repeat identical simulations to observe how changing inputs may affect system performance. That is, specifying a seed ensures differences in calculated metrics are not due to different simulation runs. If left unspecified, the simulation engine will obtain a random number seed for its calculations.

- Always use simulation: If possible, RBD calculations use analytical (non-simulated) calculation methods to obtain exact results. However, in some cases, analysts prefer to utilize the Monte Carlo simulation engine instead of analytical techniques. This selection allows the ability to choose to always run the simulator engine for RBD calculations.

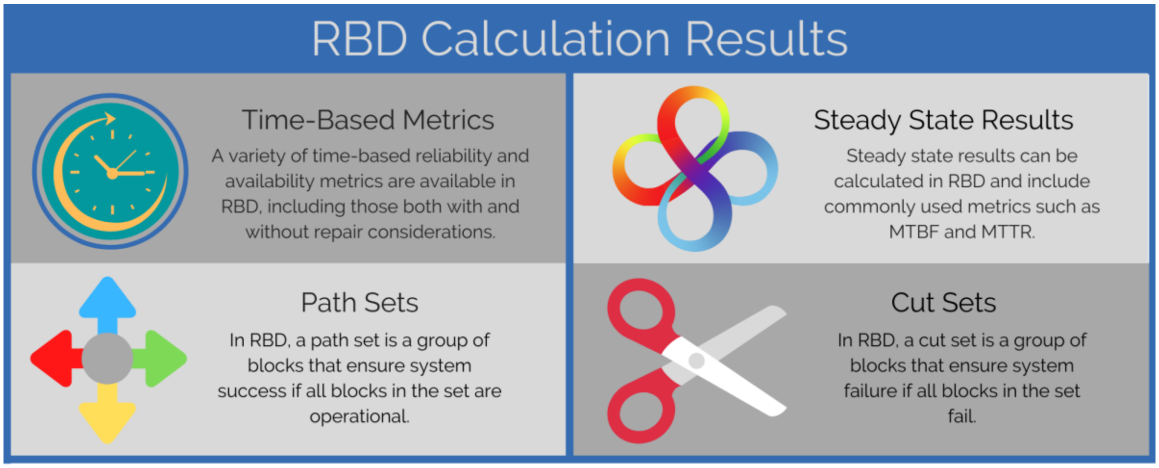
## Path Sets

In an RBD, a Path Set is a collection of components that ensure system success if all components are operational. In other words, system success is guaranteed if none of the blocks fail in a given path set. They can be thought of as the ways to connect the start and end nodes through the diagram's blocks. A variation on Path Sets, called minimal Path Sets, is a set such that, if one block is removed, the remaining blocks can no longer be considered a path set.

## Cut Sets

In an RBD, a Cut Set can be thought of as the opposite of a Path Set. A Cut Set is a collection of blocks that ensure system failure if all blocks in the set fail. Contrary to path sets, they can be thought of as the ways to break the connection between the start and end node in a given diagram. Cut Sets can help to target specific areas for product improvement efforts by identifying critical components or groups of components that lead to system failure.

For complex systems, there can often be many thousands of Cut Sets. For this reason, the concept of minimal cut sets is important to narrow the scope of analysis. Similar to minimal path sets, a minimal cut set is a set such that, if one block is removed, the remaining blocks can no longer be consider a cut set on their own. For a given system, each minimal cut set can be thought of as an individual failure mode. To narrow the scope of Cut Set analysis and ensure that only the most important component groups are considered, RBD analysis generally provides cut-offs for the order (the number of components in a set) and set availability for Cut Set calculations.



# RBD as Part of Your Reliability Analysis Platform

While powerful as a standalone tool, RBD analysis can become even more effective through its integration with other RAMS analysis tools. Relyence RBD is part of the integrated Relyence platform, so Relyence RBDs can access metrics from your other Relyence products. Blocks can be linked to subsystems in Relyence Reliability Prediction or Weibull data sets, so component failure behavior can be automatically obtained from other reliability analyses. By integrating RBD with other reliability analysis tools, you leverage related component data to ensure your system model and results are accurate. In addition, product integration can help to save time during diagram creation and component definition.

## Leveraging Failure Data in RBD

### Reliability Prediction and RBD

Reliability Prediction analysis is a powerful MTBF assessment tool that utilizes the parts in a system's BOM (Bill of Materials) to perform its calculations. At first, it may seem as though both RBD and Reliability Prediction analyses perform the same function of calculating a failure rate and MTBF. However, the key difference is that RBD analysis can take into account more complex system configurations, such as those that contain redundant component and switching characteristics, in its computations.

By integrating these two powerful RAMS analysis tools, you leverage the granular part-level data from Reliability Prediction with the redundancy support and full range of reliability and availability calculations from RBD. In Relyence RBD, blocks can be linked to Reliability Prediction subsystems or parts to automatically bring in the associated record's MTBF for RBD calculations.

Below are two examples of how these tools can work in tandem and improve the effectiveness of each other. Both are based on our RC Helicopter example findings from the "Review and Evaluate" section in our blog post *A Guide to System Modeling Using RBD Analysis*.

- Our first suggestion to bring our system performance into organization compliance standards was to improve the Flight Controller subsystem from 20 FPMH (Failures per Million Hours) to 8 FPMH. What we didn't discuss was how to achieve such drastic improvement! The PCB (printed circuit board) for our Flight Controller is made up of a number of components, including integrated circuits and passive components (resistors, capacitors, inductors, etc.). Reliability Prediction provides an efficient platform to define these components and calculate the resulting subsystem performance. This allows for efficient and effective testing and evaluation of alternate PCB parts for the Flight Controller assembly. If the Flight Controller block in RBD is linked back to the Flight Controller subsystem from Reliability Prediction, updated performance properties based on other design alternatives will be automatically included in RBD.
- Our second suggestion to improve the performance of our RC helicopter was to add a backup battery unit. The advantages of integration in this case are apparent! In Reliability Prediction, the components and parts that make up the battery assembly can be defined. Then in RBD, it's as simple as adding and defining the appropriate redundant configuration (in this case, a 1-out-of-2 Cold Standby setup) and linking each block to the battery assembly in Reliability Prediction.

## Weibull and RBD

Weibull analysis, also known as Life Data analysis, is a RAMS tool that uses mathematical techniques to fit field failure data to statistical distributions. This information can be used to evaluate product performance and predict future trends.

By integrating Weibull with RBD, you get an accurate model of your system that reflects real-life performance. Using statistical distributions to model component performance also allows for more complex failure behavior to be included in your analysis. Components with early life or age wear-out related failures can be accurately modeled using Weibull statistical techniques and included in RBD calculations.

In Relyence RBD, blocks can be linked to Weibull data sets to automatically bring in life data trends for the most efficient and effective reliability and availability calculations.

## Leveraging Repair Data in RBD

### Maintainability Prediction and RBD

Maintainability Prediction analysis based on the MIL-HDBK-472 standard provides a breakdown of the maintenance tasks required to repair a component or system. After maintenance tasks are defined for a given subsystem, detailed maintenance metrics including MTTR can be calculated.

By using Maintainability Prediction in conjunction with RBD, you can ensure you are getting the most accurate predictive model for your repairable components. By providing a granular framework to delineate each repair task along with its associated maintenance time, Maintainability Prediction makes sure all appropriate repair procedures are included. In addition, the ability to configure Maintenance Groups and Maintenance Task Groups allows for an organized and efficient repairable component strategy.

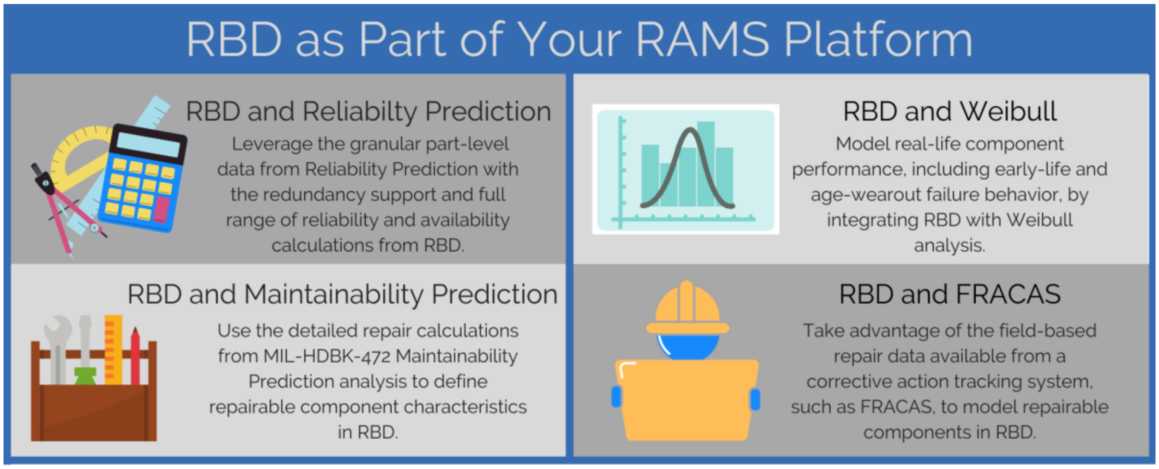
With both Maintainability Prediction and RBD available as tools within the Relyence Studio reliability platform, calculation results from MIL-HDBK-472 Maintainability Prediction calculations can be used to define repairable component characteristics in to ensure system modeling accuracy.

### FRACAS and RBD

A corrective action process system, such as FRACAS, is a powerful tool that can be used to track field failure and repair activities. With the ability to proceed through defined workflow steps from initial report to incident closeout, FRACAS can provide helpful insight into real-life repair intervals.

Similar to the integration between Weibull and RBD, using FRACAS with RBD means that you will have accurate repair models of your system's components that mirror real-life performance. By using MTTR values that are calculated based on field maintenance activities, you ensure that any supplementary repair aspects, such as logistic and other delays, are taken into account in RBD calculations.

With both FRACAS and RBD available as tools within the Relyence Studio reliability platform, field-based MTTR calculations from FRACAS can be used in RBD to ensure repairable component models reflect real-world performance.




# Using RBD Dashboards for High-Level Overviews

To further enhance reliability analyses, calculated results from RBD can be combined on a dashboard to view high level trends. While each individual RBD can provide its own benefits from analysis, the results can be even more powerful when data is combined from multiple RBD analyses. By having reliability and availability metrics easily available across product lines, manufacturing locations, etc., specific products or product areas can be identified to best target improvement efforts. In addition, if similar parts are used across product lines, high-level overviews can help pinpoint the most problematic individual components for improvement or added redundancy.

The Relyence RBD module is coupled with the Relyence RBD Dashboard to provide an at-a-glance overview of reliability and availability measures from your block diagram analyses. The Relyence RBD Dashboard can be completely customized to your needs. You can select the widgets you want to see on your dashboard, as well as what systems you want to monitor, along with various other data filtering options.

## Relyence RBD Dashboard Widgets

The Relyence RBD Dashboard includes these widgets:

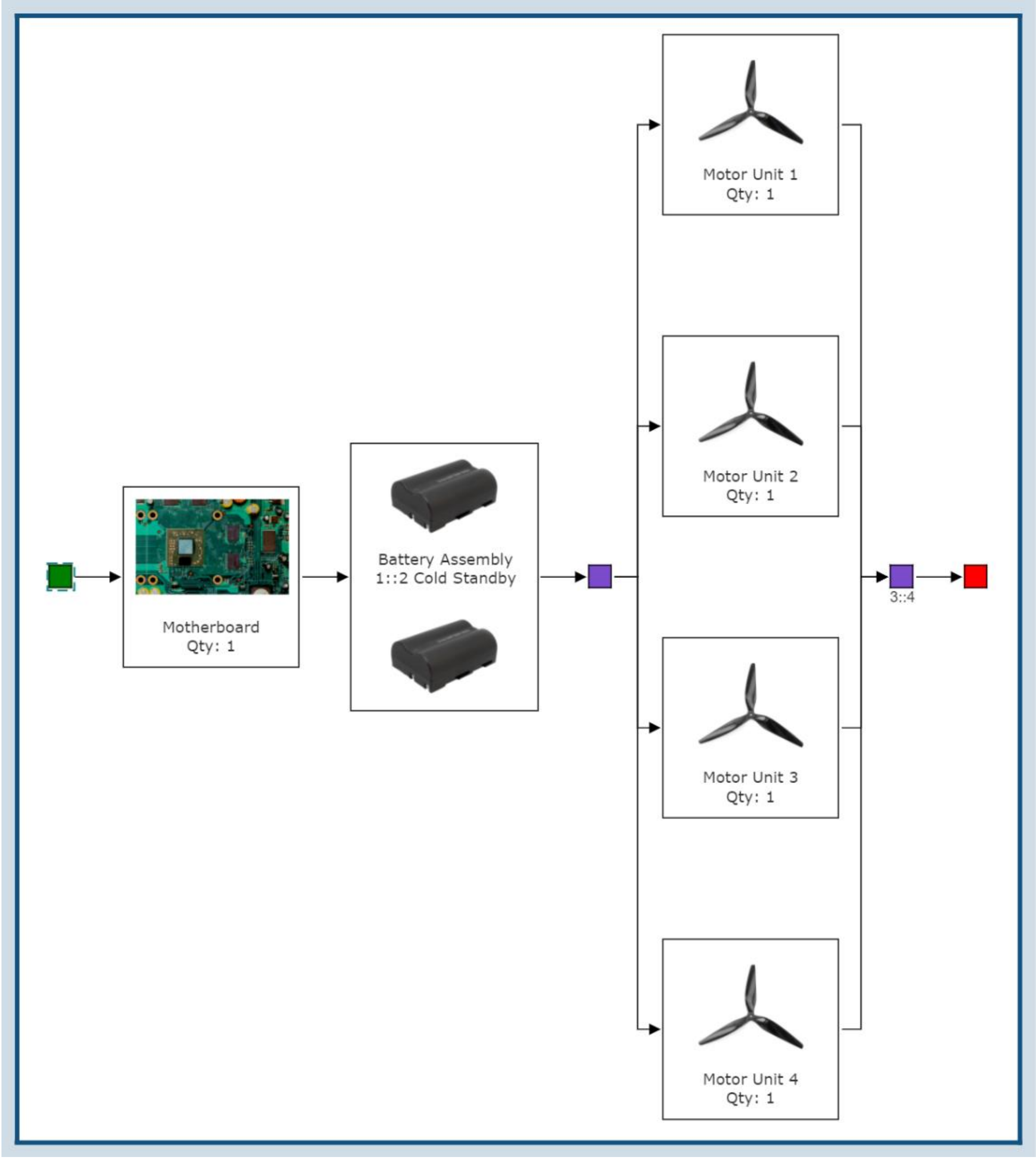
- Blocks Linked to Components: A listing of diagram blocks that are linked either to Reliability Prediction subsystems or parts.

- Blocks with High Results: A listing of diagram blocks that are greater than a specified result value. You can select the result value you want to filter on and the result value above which you want to see.
- Blocks with Low Results: A listing of diagram blocks that are less than a specified result value. You can select the result value you want to filter on and the result value below which you want to see.
- RBD Path Sets: A listing of the Path Sets of your RBD(s).
- RBD Cut Sets: A listing of the Cut Sets of your RBD(s).
- Repairable Blocks: A listing of diagram blocks that are designated as repairable in your RBD(s).
- Tagged Blocks: A listing of diagram blocks that are designated as tagged items in your RBD(s).
- Top N Unavailability Blocks: A listing of the top N number of diagram blocks with the highest or lowest unavailability. Relyence allows you to customize the number of block results that appear on this widget.
- Top N Unreliability Blocks: A listing of the top N number of diagram blocks with the highest or lowest unreliability. Relyence allows you to customize the number of block results that appear on this widget.

The Relyence RBD dashboard widgets can be used in combination with other Relyence product widgets if you are using more than one product in the Relyence line. Using cross-module dashboards can provide you with a powerful mechanism for keeping track of all your reliability metrics and help you better attain and maintain your reliability and quality objectives.

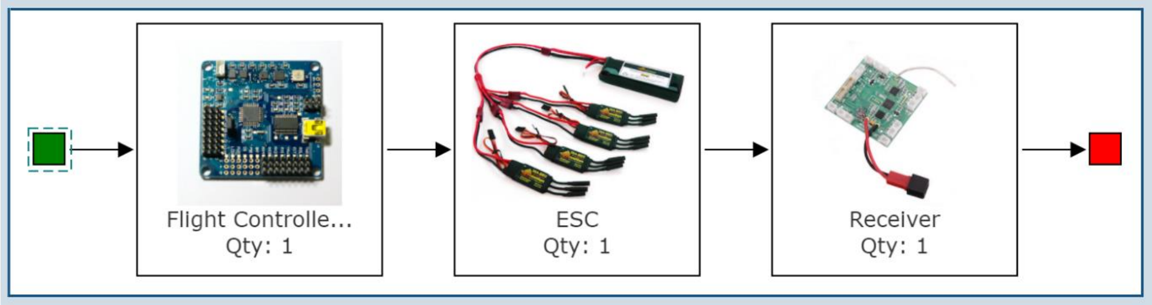
# Example RBD Analysis

To better understand the concept of RBD calculation results, a basic example RBD of a Quadcopter Drone will be analyzed. Its general structure is shown below:



Quadcopter Drone RBD block components are defined as:

- Motherboard assembly – Set up as a subdiagram under the top-level RBD. In the subdiagram RBD, the Motherboard consists of a Flight Controller, Electronic Speed Controller (ESC), and Receiver connected in series with each other. Subdiagram RBD is shown below:




  Failure characteristics for each block are defined as:

  - Flight Controller PCB: Failure Rate = 20 FPMH (Failures per Million Hours)
  - Electronic Speed Controller: Failure Rate = 10 FPMH
  - Receiver: Failure Rate = 10 FPMH

- A redundant setup of 2 Battery Assemblies, only one of which needs to be operational in order for the Quadcopter Drone to function. Each battery unit has an MTBF of 15,000 hours. Redundancy type is Cold Standby with Switch Probability equal to 1 and Switch Delay of 0 minutes.
- The 4 Motor Units - for this Parallel redundant setup, 3 of the 4 Motor Units must be operational for the Quadcopter Drone to function and Switch Probability equals 1. The failure rate of each Motor Unit is 5 FPMH.
- The Motherboard Assembly, the 2 redundant Battery Assemblies, and the 4 redundant Motor Units are connected in series with each other.

For this Quadcopter Drone system, all previously defined time-based and steady state metrics were selected for calculation. In addition, the following calculation inputs were set:

- End time = 1000 hours
- Number of display steps = 10
- Number of iterations = 10,000
- Number of failures to reach steady state = 1
- Set random number seed = Not specified
- Always use simulation = Not selected

Time-based and steady state calculation results from Relyence are below. Note that, in this case, Relyence was able to solve this RBD analytically.

### Quadcopter Drone Results

| | | | |
|---|---|---|---|
| **Diagram Name** | Quadcopter Drone | **MTTF** | 14741.750997 |
| **Calculation Method** | Analytical | **MTBF** | ∞ |
| | | **MTTR** | ∞ |
| | | **Steady State Availability** | 0.000000 |

TIME-BASED RESULTS

| Time | Reliability | Unreliability | Failure Rate | Equivalent Failure Rate | Availability | Unavailability | Mean Availability | Mean Unavailability | Hazard Rate | Failure Frequency | Total Downtime | Expected Number of Failures |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1.000000 | 0.000000 | 40.000000 | 40.000000 | 1.000000 | 0.000000 | 1.000000 | 0.000000 | 40.000000 | 40.000000 | 0.000000 | 0.000000 |
| 100 | 0.995984 | 0.004016 | 40.471449 | 40.236222 | 0.995984 | 0.004016 | 0.997995 | 0.002005 | 40.471449 | 40.308934 | 0.200527 | 0.004016 |
| 200 | 0.991939 | 0.008061 | 40.936984 | 40.470463 | 0.991939 | 0.008061 | 0.995979 | 0.004021 | 40.936984 | 40.606973 | 0.804134 | 0.008061 |
| 300 | 0.987863 | 0.012137 | 41.396719 | 40.702752 | 0.987863 | 0.012137 | 0.993954 | 0.006046 | 41.396719 | 40.894305 | 1.813802 | 0.012137 |
| 400 | 0.983760 | 0.016240 | 41.850768 | 40.933117 | 0.983760 | 0.016240 | 0.991919 | 0.008081 | 41.850768 | 41.171114 | 3.232404 | 0.016240 |
| 500 | 0.979630 | 0.020370 | 42.299236 | 41.161586 | 0.979630 | 0.020370 | 0.989875 | 0.010125 | 42.299236 | 41.437582 | 5.062708 | 0.020370 |
| 600 | 0.975473 | 0.024527 | 42.742232 | 41.388186 | 0.975473 | 0.024527 | 0.987821 | 0.012179 | 42.742232 | 41.693889 | 7.307380 | 0.024527 |
| 700 | 0.971291 | 0.028709 | 43.179858 | 41.612944 | 0.971291 | 0.028709 | 0.985759 | 0.014241 | 43.179858 | 41.940211 | 9.968983 | 0.028709 |
| 800 | 0.967085 | 0.032915 | 43.612214 | 41.835885 | 0.967085 | 0.032915 | 0.983688 | 0.016312 | 43.612214 | 42.176725 | 13.049979 | 0.032915 |
| 900 | 0.962856 | 0.037144 | 44.039398 | 42.057034 | 0.962856 | 0.037144 | 0.981608 | 0.018392 | 44.039398 | 42.403602 | 16.552735 | 0.037144 |
| 1000 | 0.958605 | 0.041395 | 44.461506 | 42.276418 | 0.958605 | 0.041395 | 0.979520 | 0.020480 | 44.461506 | 42.621012 | 20.479518 | 0.041395 |

In addition to time-based and steady state metrics, we also want to examine the path sets and cut sets for the Quadcopter Drone to identify critical components. As can be seen in the diagram, system operation will be successful if the Motherboard Assembly, the redundant Battery Assembly system, and 3 of the Motor Units are operational.

Therefore, the full list of path sets is:

- Flight Controller > ESC > Receiver > Battery Assembly > Motor Unit 1, Motor Unit 2, Motor Unit 3
- Flight Controller > ESC > Receiver > Battery Assembly > Motor Unit 1, Motor Unit 2, Motor Unit 4
- Flight Controller > ESC > Receiver > Battery Assembly > Motor Unit 1, Motor Unit 3, Motor Unit 4
- Flight Controller > ESC > Receiver > Battery Assembly > Motor Unit 2, Motor Unit 3, Motor Unit 4

For this RBD, the list of cut sets includes any subcomponent of the Motherboard subdiagram, the redundant Battery Assembly system, and 2 of the Motor Units. In other words, cut sets include:

- Flight Controller
- ESC
- Receiver
- Battery Assembly
- Motor Unit 1, Motor Unit 2
- Motor Unit 1, Motor Unit 3
- Motor Unit 1, Motor Unit 4
- Motor Unit 2, Motor Unit 3
- Motor Unit 2, Motor Unit 4
- Motor Unit 3, Motor Unit 4

Note that the cut sets above are the minimal cut sets of the example system. The list of all cut sets would include any of the above Minimal Cut Set groups and any additional blocks in the system. For Example, a Cut Set could be comprised of the Flight Controller, ESC, and Receiver. The system will fail if all 3 of these blocks fail, although it is not a Minimal Cut Set because each block on its own will cause system failure.

Path Set and Cut Set calculation results from Relyence are below:

PATH SETS

| Availability | Blocks |
|---|---|
| 0.944473 | Flight Controller, ESC, Receiver, Battery Assembly, Motor Unit 2, Motor Unit 3, Motor Unit 4 |
| 0.944473 | Flight Controller, ESC, Receiver, Battery Assembly, Motor Unit 1, Motor Unit 3, Motor Unit 4 |
| 0.944473 | Flight Controller, ESC, Receiver, Battery Assembly, Motor Unit 1, Motor Unit 2, Motor Unit 4 |
| 0.944473 | Flight Controller, ESC, Receiver, Battery Assembly, Motor Unit 1, Motor Unit 2, Motor Unit 3 |

Cut Sets

| Availability | Blocks |
|---|---|
| 0.980199 | Flight Controller |
| 0.990050 | ESC |
| 0.990050 | Receiver |
| 0.997874 | Battery Assembly |
| 0.999975 | Motor Unit 1, Motor Unit 2 |
| 0.999975 | Motor Unit 1, Motor Unit 3 |
| 0.999975 | Motor Unit 1, Motor Unit 4 |
| 0.999975 | Motor Unit 2, Motor Unit 3 |
| 0.999975 | Motor Unit 2, Motor Unit 4 |
| 0.999975 | Motor Unit 3, Motor Unit 4 |

## Conclusion

Given its complexity and assortment of modeling options, it can often be difficult to get started with RBD. However, once understood, it is one of the most valuable RAMS tools available to model and evaluate system performance. By incorporating numerous component behavior models, supporting redundant configurations, and including a powerful calculation engine with a variety of reliability and availability metric options, RBD ensures the results you obtain are effective and accurate.

Our mission at Relyence is to build not only the most capable tools, but also the most technologically advanced and well-crafted applications available. We rely on our industry experience and expertise to build the tools reliability experts expect with a design elegance and utility that makes our tool suite stand out.

Relyence RBD accomplishes this goal on many fronts. Our front-end diagramming tool is easy to use with built-in smart layout and allows you to create effective and visually-pleasing RBDs. Our back-end calculation engine can efficiently compute a broad range of time-based and steady state metrics. Added to those core capabilities is a powerful set of features to extend RBD analysis to a new level, including:

- Dashboards to view high level metrics
- Subdiagram pane to display hierarchical subsystem relationships and efficiently manage large systems
- Integration with Relyence Reliability Prediction and Relyence Weibull
- Flexible report generation
- Role-based permissions and access control capabilities

With its extensive list of features, Relyence RBD stands out as a top performer in system modeling packages. To learn more about the advantages of Relyence RBD, or to see it in action, sign up today for your own no-hassle free trial. Or, feel free to contact us to discuss your needs or schedule a personal demo.