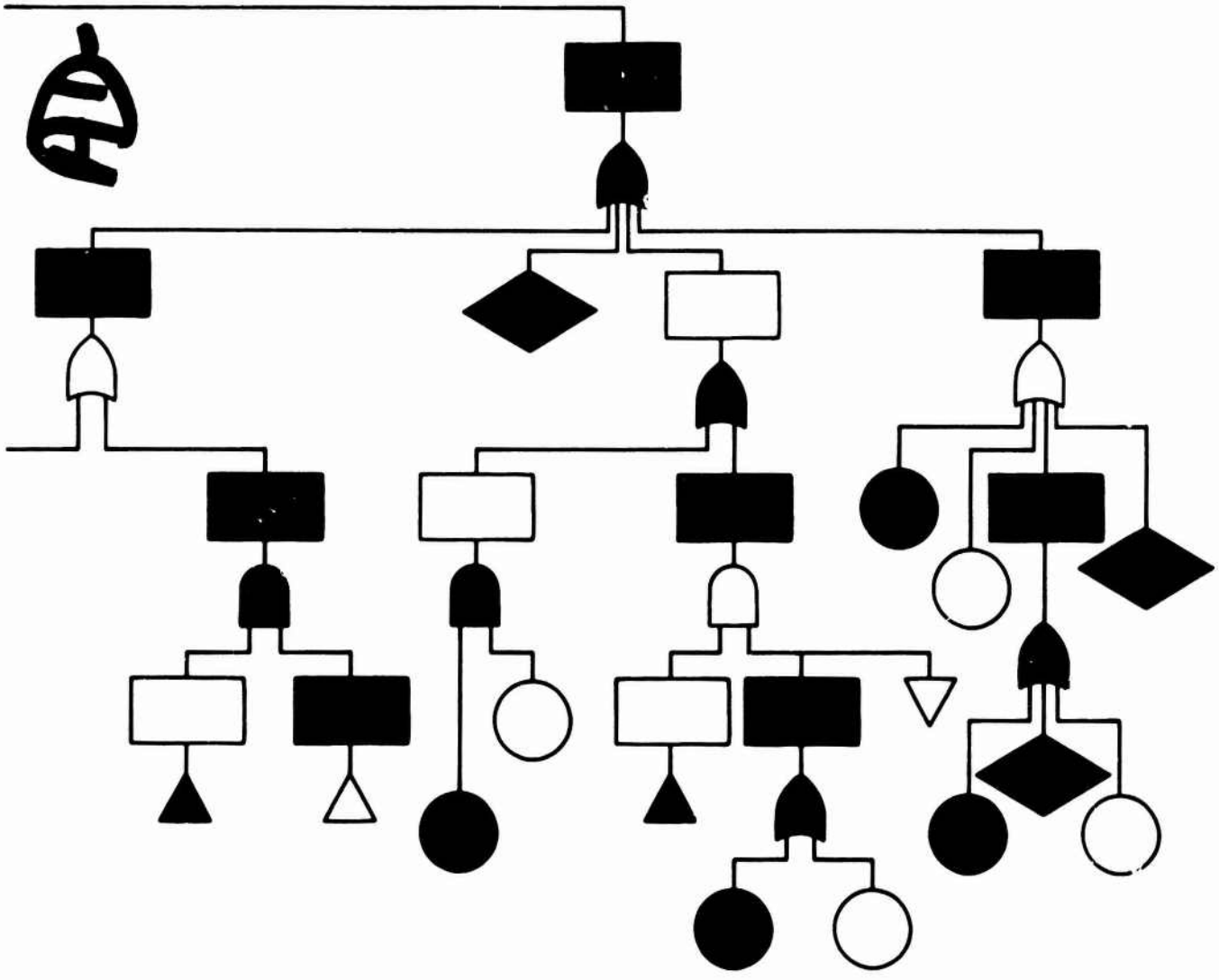
847015

L

BOEING AEROSPACE COMPANY
RESEARCH AND ENGINEERING DIVISION
SEATTLE, WASHINGTON

FAULT TREE FOR SAFETY

AD



SUPPORT SYSTEMS ENGINEERING

PURPOSE

This document has been prepared to acquaint system managers, designers, and support engineers with the Fault Tree analysis technique employed by The Boeing Company in system satety engineering of new aircraft designs.

Use of this accident prevention tool is one of the means being utilized to help ensure that next generation military and commercial aircraft meet the higher safety levels required for future operations.

This document updates and suspersedes Boeing document D6-57133, same title, dated January 1966.

INTRODUCTION

The Fault Tree analysis technique was employed successfully for the first time in the Minuteman ICBM Program. Guided by the critical fault paths identified by fault tree systems analysis, design engineers were able to eliminate or control many hazards before accidents occurred. As a result, the USAF rates Minuteman as the "safest" ICBM in the inventory. More recently, the Fault Tree technique has been applied by Boeing to the AGM-69 (SRAM) program.

With this experience as a baseline, Boeing engineers and analytical specialists have further refined the Fault Tree analysis technique and adapted it to aircraft design. The resulting technique has been used to advantage on the Supersonic Transport (SST) and other recent commercial transport programs.

This document presents the philosophy and application of Fault Tree analysis as employed by The Boeing Company in aircraft system design. The main body of the publication describes the basic concept of Fault Tree analysis, introduces the method for diagramming events to be analyzed, and develops a sample problem from conceptual development through analysis and rework of the results.

SUMMARY

Effective system safety engineering requires a method for examining proposed designs, identifying potential undesirable events, and recommending solutions that will prevent those events from occurring. To accomplish this, the Fault Tree technique was conceived by the Bell Telephone Laboratories in 1962. Subsequently, Boeing successfully applied the technique to the Minuteman ICBM system. Recent refinement of the technique has permitted its adaptation to dynamic systems such as aircraft. The same desirable features that gained Fault Tree its wide acclaim on Minuteman has been retained in its adaptation to aircraft systems.

The Fault Tree process utilizes logic diagrams to portray and analyze potentially hazardous events. As employed by Boeing, this involves the following six steps:

1. Define undesired event
2. Acquire understanding of system
3. Construct fault tree
4. Collect quantitative data
5. Evaluate fault tree probability
6. Analyze computer results

Three basic symbols (logic gates) are adequate for diagramming any fault tree. However, additional recently developed symbols can be used to reduce the time and effort required for analysis. In addition, use of a new technique, called "Importance Sampling" for generating failure occurrences serves to dramatically reduce the amount of computer time required to produce quantitative results.

Fault Tree analysis, as currently developed, can be applied to virtually any system, design, or procedure with positive results. It fills the need for a quantitative safety analysis capable of extreme detail. In addition, it provides an extremely useful tool in determining the weak points in a design, whether or not numerical analysis is applied.

TABLE OF CONTENTS

LIST OF FIGURES

NEED FOR SYSTEM SAFETY ANALYSIS

The goal of system safety engineering is to identify, evaluate, and eliminate or control potential hazards as early in the system life-cycle as possible. This goal can best be achieved by analyzing the design, development, fabrication, test, installation, maintenance, and operation of complex man-machine systems.

The system safety engineering concept, an extension of the traditional accident prevention program, utilizes trained safety engineers to analyze and evaluate the total system. The concept involves a minute examination of each design, including application of computer techniques, to evaluate failure effects, man-machine relationships, and all aspects of system development and operation. Finally, solutions are recommended to the decision-making authorities.

The advantage of using a quantitative basis for decision-making is readily apparent; the method used to quantify safety is not. To illustrate the scope of safety analysis as opposed to reliability analysis, consider a leak in an engine fuel line. A reliability analysis would normally consider only the immediate effect (i.e., is the leak sufficient to cause the engine to operate out of tolerance?). In the evaluation of system safety, the effect of the malfunction becomes more complex. The following significant variables have to be considered in the analysis:

1. The amount of power degradation caused by the loss of fuel
2. The operating circumstances under which the loss of power occurred
3. The effects of the degraded, or lost, engine-driven accessories
4. Asymmetric power characteristics
5. Effects of range performance caused by loss of power
6. Fire or explosion that could result from fuel leakage

The foregoing factors could result from a single component failure. Further, the effects of the time that the failure existed in the system would have to be considered. The loss of engine power might continue throughout the time period; however, the leak could be stopped promptly by actuating shutoff valves or minimized by pump deactivation. Another factor that complicates the analysis is that the planned operating period could be reduced by initiating abort action.

From the above, it is evident that the systems safety analyst cannot be satisfied with reliability analyses alone. The scope of the analysis must be greater. For this reason, it is necessary to consider factors other than materiel failures or malfunctions. In attempting to assess potential failures, the system safety engineer must consider interactions within a system and the effects of failures in many areas. Obviously, an all-encompassing analytical technique, supported by computer programming has to be developed to answer the question: "How safe. . .?"

THE FAULT TREE

A fault tree is a graphical representation of the relationship between certain specific events and the ultimate undesired event. Development of the fault tree analysis technique for system safety began in 1962 at the Bell Telephone Laboratories. The initial program was successfully applied to the Minuteman ICBM.

Boeing further developed the fault tree technique to permit its application to manned aircraft design. When employed in the design of a dynamic system, fault tree provides the systems safety engineer with the necessary information to identify and evaluate potential hazards.

The following six steps are required in fault tree analysis:

Step 1–Define undesired event
Step 2–Acquire understanding of system
Step 3–Construct fault tree
Step 4–Collect quantitative data
Step 5–Evaluate fault tree probability
Step 6–Analyze computer results

STEP 1–DEFINE UNDESIRED EVENT

To measure the level of safety of an operational product, the initial step must be definition of the most undesired event, i.e., the event that must be kept from happening.

Definition of the most undesired event is not always as simple as it might appear from a superficial view of any system, be it a power lawnmower or a supersonic aircraft. Injury to the operator may appear to be the most undesired event to the lawnmower manufacturer. Loss of life and destruction of the airplane could well be selected as the most undesired event for the supersonic plane designer. However, both of these obvious selections may be inappropriate.

Because of possible lawsuits, the lawnmower manufacturer might be concerned with the remote possibility of his product throwing a blade and fatally injuring a bystander or passerby. Implausible? Statistically, it could happen. In practice, it is more likely that the undesired event would be "loss of blade retaining nut" or something equally probable.

In the case of the supersonic transport, the undesired event could be overstressing the airframe at high Mach number or failure to arrest the rate of sink of ILS approach. True, either example could lead to destruction of the aircraft, but these are conditional situations that have to be considered and avoided. Therefore, the undesired event may not always be the direct result of a malfunction or incorrect procedure but, rather the final single action that must be avoided.

The fault tree, since it is single-event oriented, must be constructed with only one "most undesired event." There will probably be several events that lead to the "top" event and as such, they are analyzed in relationship to the top event. This situation makes it mandatory to establish terminology for the top event that will encompass the lesser events, individually or collectively.

STEP 2–ACQUIRE UNDERSTANDING OF SYSTEM

The safety of any system must be measured for a specific time and type of activity. For this reason, the system safety engineer must understand the system and its intended use.

One objective of the analyst is to determine how the system, including the people involved with system operation and maintenance, could fail and cause the undesired event.

Another objective, although no less important, is for the analyst to acquire an understanding of how the system will be used. This includes factors such as: duration and type of flights, stress levels anticipated, maintenance concept, command and control, anticipated experience level of operating and maintenance personnel, etc. Also required is a knowledge of planned emergency and abort procedures so that the possibility of failure to react, or incorrect reaction of the crew, can be considered.

To analyze the whole system requires a diverse group of engineers. This requirement can be met by utilizing experts from other engineering disciplines as consultants to the fault tree team.

STEP 3–CONSTRUCT FAULT TREE

A fault tree is constructed by properly relating all possible sequences of events that, upon occurrence, result in the undesired event.

Beginning with the most undesired (top) event, the fault tree graphically depicts the paths that lead to each succeeding lower level of the display. This does not imply that each descending fault path has a "higher probability of occurrence"; in fact, in many instances, the opposite may be the case. However, a series of "little things," each with a relatively low probability of occurrence, may trigger an event at the next higher level. This is depicted in the fault tree as a progression of events through the logic gates.

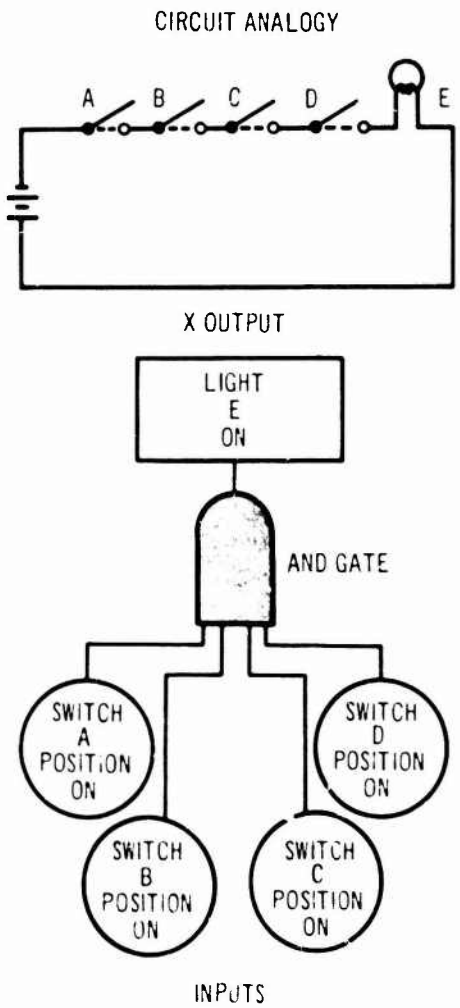
For example: A failed antiskid unit combined with a slippery runway and a severe crosswind could logically lead to divergence off the runway upon landing. If we carry this fault path higher in the display, we may find that a failed engine prohibits correcting the divergence. In this case, the multiple factors did not cause the engine to fail, but the fact that it did fail at a critical moment prevented the pilot from completing corrective action. Suppose, however, the engine failed prior to touchdown. Obviously, the pilot would have planned his approach to compensate for the power loss. Certainly, he would have been more cautious of the slippery runway and, as a consequence, better prepared to cope with the failed antiskid at the first indication of failure or malfunction. Thus, the fault tree analyst must foresee not only grossly probable events but many possible events.

BASIC LOGIC GATES

Three basic symbols, or logic gates, are used in constructing a fault tree: the AND, the OR, and the INHIBIT gates. These are illustrated in Figs. 1, 2, and 3.

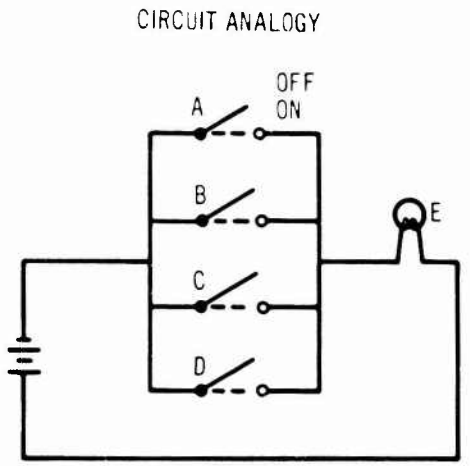
AND and OR Gates

These gates represent the fundamental Boolean functions that form the basis for all logic analysis. The decision on which gate, the AND or OR, to use can be explained by the following

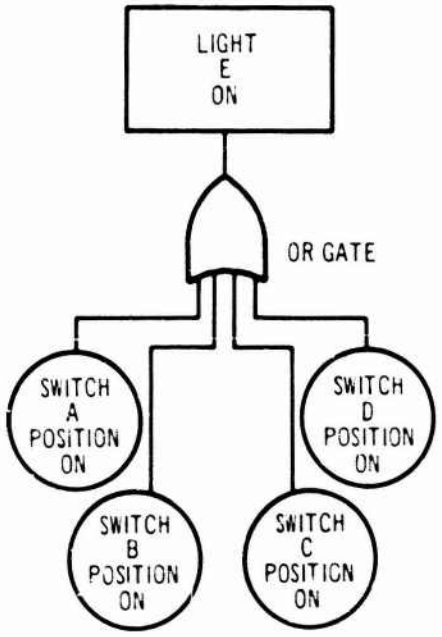


The AND gate performs the logic function that requires the co-existence of all gate inputs (A, B, C, D) events in order to realize an output (X) event.

Figure 1. Use of "AND" Gate

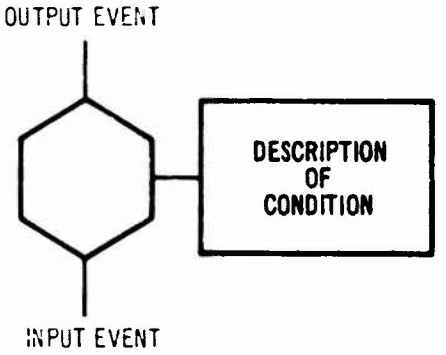


X OUTPUT

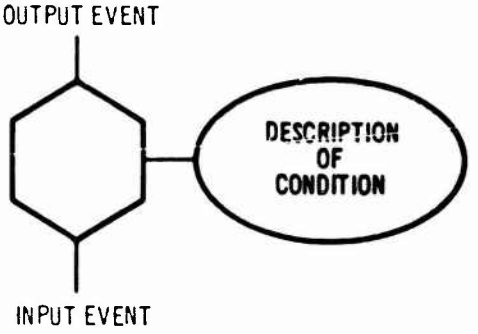


The OR gate performs the logic function that requires any one of the gate input (A, B, C, D) events in order to realize an output (X) event.

Figure 2. Use of "OR" Gate



The INHIBIT gate provides a means of applying conditional probabilities to the fault sequence. If the input event occurs and the condition is satisfied, an output event will be generated.



The RANDOM INHIBIT gate is functionally the same as the INHIBIT gate. However, in this case the conditional input is a variable.

Figure 3. Use of "INHIBIT" Gate

simple rule: If the event being considered will, by itself, cause the next higher event to occur, use an OR gate. Otherwise, determine what is necessary and sufficient to cause the next higher event and use an AND gate.

INHIBIT Gate.

The third gate, the INHIBIT gate, is a variation of the AND gate. Use of the INHIBIT gate enables the analyst to apply conditional probabilities to a fault sequence. A sample situation is that of a directional control system malfunction inducing a yaw moment during takeoff. Here, the INHIBIT gate would be flagged as "yaw moment beyond aircraft directional control capability." The output event would be "divergence off the runway." In some situations, a malfunction in the directional control system that induces a yaw moment will not result in divergence off the runway. The unprogrammed yaw moment is inhibited with the probability that the yaw moment will be beyond the aircraft directional control capability. Thus, the probability of divergence off the runway is closer to the true value.

STEP 4—COLLECT QUANTITATIVE DATA

Statistical data based on experience are used in developing probability factors. Events, as defined in the fault tree, have a particular probability of occurrence that determines system safety. Therefore, if the system operation is to be truly a "calculated risk," as opposed to a "gamble," the numerical probability of event occurrences must be determined.

There are two basic techniques for obtaining probability input data. In the first method, a specific event is tested a number of times and a statistical analysis is made of the resulting failures and successes. This approach is often the best for:

- Small component failures
- Possible human errors
- Human performance capability or limitations.

In the second method, the probability is calculated from the variables that could cause the undesired event to occur. In this approach, the probability of the variables must first be known. Such an approach is often the best for determining:

- Large component failures
- System performance capability
- Environmental conditions

SMALL COMPONENT FAILURE

Small component failures comprise the major portion of fault tree input data. In construction of a fault tree, a concerted effort should be made to continue the analysis down to relevant component failure modes.

Failure rate data are derived from operational experience or laboratory tests. These data may be expressed as operating time-per-unit failure or cycles-per-unit failure. The probability that an event will occur in a specified period of time can be calculated when this failure rate is known.

The duration of a component failure is referred to as "effective fault duration." This factor is of extreme importance to the analyst. While a failure in a redundant system may not, in itself, cause the undesired event, the length of time the failure exists determines the amount of safety degradation incurred until the failure is detected and corrected. Where mission scheduling must be predicted on a system that is 100-percent operable, an undetected component failure reduces the probability of mission success to an unacceptable degree.

Additionally, a component failure that may not, in itself, be of particular significance can pyramid to catastrophic proportions when compounded with other failures in the same

system. As an example, consider the following:

A system operates with a safety monitor feature. If Component A fails, the failure is detected by the safety monitor and any disastrous effects are inhibited. However, if Component A fails and the safety monitor has failed, the undesired event occurs.

The probability of a sequence of events occurring in a specified time (mission length) can be calculated when failure rates and effective fault durations are known or can be estimated.

The failure rate of a component is not a constant value throughout a mission, but varies with the type of activity being performed or stress levels encountered. For example, an engine has a higher failure rate at maximum power than at idle. Therefore, the failure rate for a particular engine will be proportional to the engine power levels used during a mission.

Additionally, certain mission requirements may involve other than normal stresses during flight, such as terrain avoidance or aerial delivery. These factors must be considered in calculating total mission safety. A typical component failure rate profile is presented in Fig. 4.

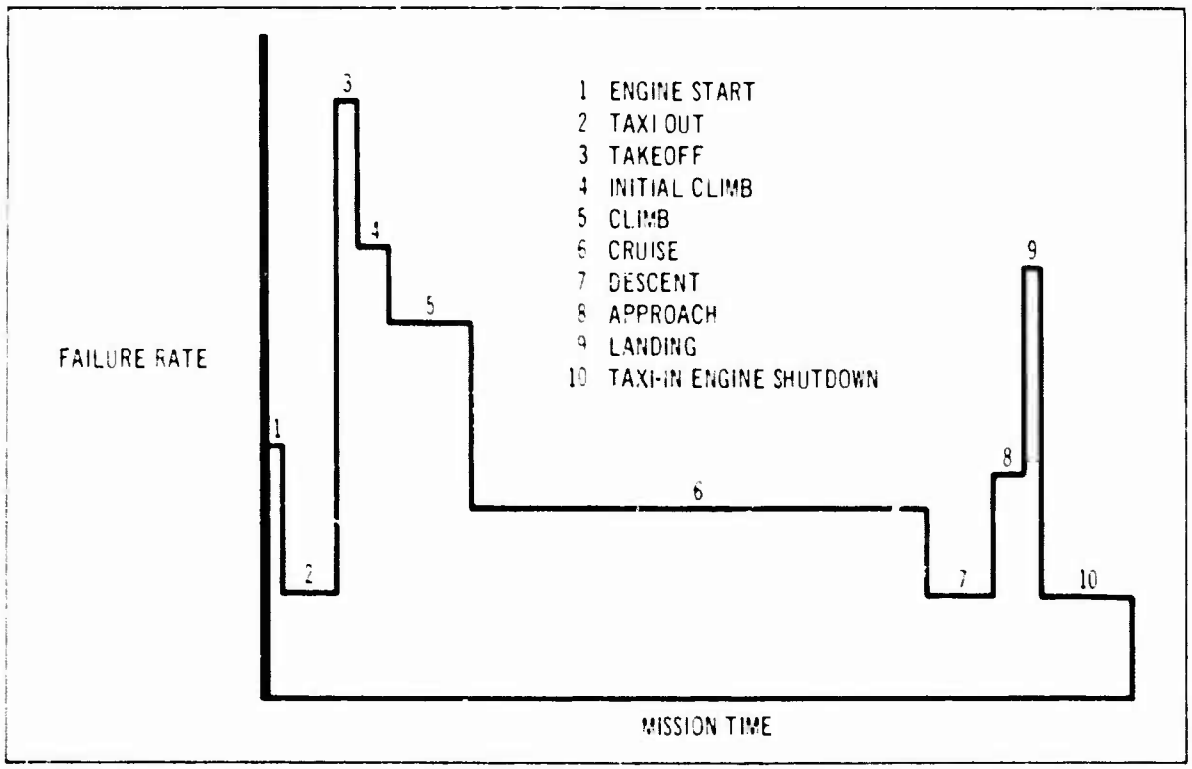


Figure 4. Typical Aircraft Component Failure Rate

Failure rate data should represent, as closely as possible, the values that may be expected during actual system operation. Data derived from operational experience will usually include those failures caused by design deficiencies, workmanship, material deficiencies, maintenance errors, etc. Since it is virtually impossible to simulate exactly the expected operational environment, the analyst should be aware of the test conditions and adjust the data accordingly.

The best sources of data that meet these requirements are operating histories of similar systems employing identical or similar components. Certain agencies have inaugurated data systems that collect and classify component data for their systems. Examples are the Navy's 3-M Maintenance and Material Management Program and the Air Force AFM 66-1 Maintenance Management Program. These programs, however, still do not make it possible to obtain complete failure data as a function of failure mode and stress level.

HUMAN PERFORMANCE

In fault tree construction and data collection, human error, both deliberate and inadvertent, must be considered. An example would be the deliberate activation of the wrong switch or the inadvertent operation of a switch at the wrong time.

Human performance capability is the probability that a crew will be able to perform a certain required function. These capabilities relate primarily to performing unscheduled functions, such as landing a multiengine aircraft in an engine-out configuration.

To secure meaningful data for fault tree analysis inputs, the flight simulator has proven most helpful. Boeing is constantly sampling representative crew capabilities. Actual conditions are duplicated as authentically as possible. However, developing numerical values for human capability has been a slow and sometimes painful process. Unlike mechanical components, the human system is far too complex to be analyzed for man-machine compatibility with 100-percent accuracy. In any event, the fault tree technique does permit inclusion of human variables that are established. We may expect an increase in analysis confidence as further progress is made in the study of human performance capability.

LARGE COMPONENT FAILURES

It is not always possible to express events related to system hardware failures in terms of components that can be tested economically to obtain valid statistical data. An example is an aircraft wing. For failures of this type, it is possible to determine the probability of an undesired event by a careful analysis of variables that could cause such an event, such as gust loads and other natural phenomena. Normally, structural failure will occur only when induced stresses are imposed that are well beyond calculated strength or fatigue levels.

ENVIRONMENTAL CONDITIONS

The term "environmental conditions" refers to events that are functions of system operating requirements. A typical environmental condition that could be encountered during operation is severe icing. Calculating the probability of occurrence of such an event is accomplished with available statistical data relative to weather environments for specific geographical areas. Advance knowledge of certain weather phenomena likely to be encountered by a particular system provides a yardstick for the designer with which he can establish design requirements compatible with the projected missions. The designer can also determine any system protection, other than standard, that may be necessary.

STEP 5—EVALUATE FAULT TREE PROBABILITY

When the fault tree has been constructed and the failure data obtained for the basic events, it is desirable to determine which combinations of failures are most likely to cause the top event, i.e., which failure paths are dominant. A secondary goal is to determine the probability of the top event occurring.

These goals might be achieved by calculating the probability for each failure path and using this information to determine the total probability. Unfortunately, there are far too many failure paths in a large fault tree to allow calculation of the probability for every path. A fault tree with 100 gates will typically have thousands of paths; a 200-gate tree, millions. Thus, even finding all the paths usually proves to be prohibitive.

FAULT TREE SIMULATION

A more practical approach would be to calculate the probability of the dominant paths only. This can be accomplished on a computer fault tree simulation program, using Monte Carlo simulation.

The fault tree simulation program requires failure data for the basic input events and a description of each fault tree gate, including definition of the output event, the type of gate, and a list of the inputs to the gate. Given the problem definition, the computer simulates failures in the fault tree, i.e., failures are randomly generated according to the failure statistics provided and, for each occurring combination of failures, the fault tree logic is used to determine whether the top event would occur. The simulation is performed until a large number of system missions have been simulated. The number of trials (mission simulations) performed is chosen so that each dominant path occurs several times. Each time the top event occurs, the failure path is traced and listed as printed output. After a sufficient number of trials are completed, the simulation statistics are used to calculate an estimate of the probability for the top event.

The Monte Carlo technique was chosen so that the frequency with which a failure combination occurs is highest for the most probable combinations. Thus, the dominant paths occur most frequently and can easily be determined by examining the printed output from the computer.

FAULT TREE IMPORTANCE SAMPLING

Early fault tree simulation programs used the most straight-forward of Monte Carlo techniques: direct simulation. That is, events were generated with frequencies equal to their natural occurrence frequencies. This technique satisfied the requirement that the most probable combinations happen most frequently, thereby emphasizing the dominant paths. However, since fault trees usually represent improbable events, the number of trials required was usually prohibitive. For example, estimation of a probability of 0.00001 would require about ten million trials to achieve a reasonable degree of confidence. A "typical" 300-gate tree requires about 1/10 second per trial, hence ten million trials would require one million seconds or about 275 hours of computer time.

In order to reduce the computer run time to an acceptable level, a Monte Carlo technique called "Importance Sampling" is used. As applied to fault tree simulation, the technique is to generate failures in a manner that increases the frequency with which the various failure combinations occur, while retaining the feature of the dominant paths occurring most frequently. The increased frequency is compensated for by the use of weighting factors inversely proportional to the increase in frequency.

Fault tree simulation with Importance Sampling usually allows evaluation with a sample size of 1,000 trials. Thus, the sample size is reduced by a factor of 10,000 for a probability of 0.00001, and more for smaller probabilities. Since the computer time per trial is increased by a factor of about 10, the net overall improvement would be a factor of about 1,000 for the above example. Thus the computer time for the above example, utilizing Importance Sampling, would be 17 minutes versus 275 hours.

After considerable application, fault tree simulation with Importance Sampling has proven to be quite successful.

STEP 6—ANALYZE COMPUTER RESULTS

As used in fault tree analysis, "system" means the entire aircraft, its crew, and support equipment. Before a total system analysis can be made, the various subsystems must be subject to initial analyses. Thus a detailed analysis of the flight control subsystem, for example, will eventually appear in a complete fault tree as one part of the total system. The objective of each analysis varies with the particular phase of system design or operation. Typical system phases are shown in Fig. 5.

The Fault Tree analysis technique may be applied advantageously to any system before design reaches the detail level. The full merits of this technique are realized, however, only after the preliminary design phase, when the system reaches the component level. Fault tree is unique in its ability to accept component-level detail without restriction.

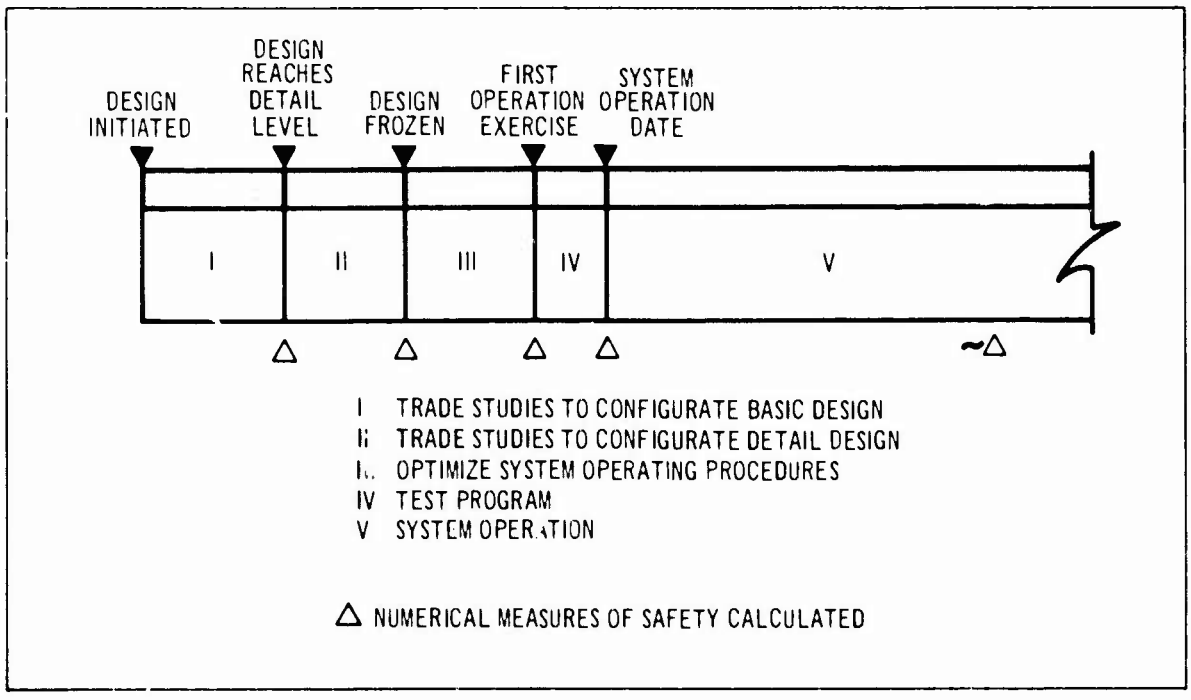


Figure 5. System Phase as Visualized by a System Analyst

Initially then, the objective of the analysis of computer responses is to obtain an indication of the measure of safety inherent in the design. Additionally, the analysis provides a baseline against which to compare various design alternatives. The area where the most significant gain in safety may likely be obtained is made apparent by the critical fault paths, identified from initial computer responses.

Seldom will a single computer run satisfy system safety requirements. For each design alternative, the effect on system safety must be measured, particularly in relation to the probability of occurrence of the undesired event. Each alternative is evaluated and the necessary changes are made to the fault tree and/or data to reflect the new design.

Finally, application of fault tree analysis results, combined with other analyses generated by design groups and support functions, assures a safe and cost-effective design.

APPLICATION OF FAULT TREE TECHNIQUE

To illustrate a typical example of fault tree application, we have selected a theoretical, multiengine aircraft. For purposes of analysis, only the takeoff phase is considered and we will assume that this is the point where both the affected design groups and the system safety engineers desire to make an initial measurement of the level of safety attained in the design of the aircraft.

STEP 1–DEFINE UNDESIRED EVENT

An undesired event was selected for analysis purposes: "Loss of Landing Gear Steering Directional Control Lost."

In actual practice, an undesired event restricted to this extent would not be acceptable as a meaningful measure of system safety, but it does serve to illustrate how the fault tree technique is applied.

The terminology used to describe various events must be compressed, but still be meaningful to the analyst. For example, if a rudder pedal "breaks," this means that the pedal failed structurally or came apart in some manner to the extent that its use was denied.

STEP 2–ACQUIRE UNDERSTANDING OF SYSTEM

System safety analysts must have a thorough knowledge of the system being analyzed. This means that they must be able to visualize all the events that could conceivably take place as a result of malfunctions or failures. Logically, such broad mental gymnastics may not be expected of one individual and, therefore, fault tree construction should be a team effort. For the purpose of this illustration, we have assumed a large aircraft with dual-wheel steering command system.

STEP 3–CONSTRUCT FAULT TREE

Figure 6 portrays a fault tree that was constructed for the undesired event defined in Step 1. The tree is read the same way it is constructed, from the top (undesired event) down. It is necessary to keep in mind the preceding event description for, as noted previously, space is limited within each diagram and repetition is usually omitted. For example, a sequence may start as an occurrence during takeoff, while, further down the tree, it will only be implied that takeoff is occurring.

STEP 4–COLLECT QUANTITATIVE DATA

Quantitative input data, representative of current aircraft systems, were obtained from many sources: The Air Force AFM 66-1 program, the Navy Maintenance and Material Management (3M) Program, and The Boeing Company experience retention system. These programs provide operational and maintenance

experience data from a number of military aircraft and virtually every airline in the free world. Input data for the fault tree are presented in Fig. 7.

STEP 5—EVALUATE FAULT TREE PROBABILITY

The probability of the undesired event occurring was determined by simulation. The probability calculation was based on one 45-second takeoff roll per 5 flight hours. The fault tree simulation consisted of 1,000 trials of 250 takeoff hours each, using Importance Sampling with an Importance Factor of 0.5.

STEP 6—ANALYZE COMPUTER RESULTS FOR SAFETY IMPROVEMENT

As shown in Fig. 8, the undesired event occurred 134 times during the 100 million hours of simulation representing 1,000 trials of 100,000 hours each. This equates to a probability of occurrence of the undesired event of 0.012/100,000 hours. This is not an unrealistic rate. It can, however, be improved significantly.

Analysis of the initial simulation determines that critical fault paths and responsible events were as follows:

1. Parts in vicinity of rudder-pedal linkage come free and restrict rudder-pedal-linkage movement (Item Z8)
2. Left metering valve fails, jams, and prevents steering system movement (Item Z5')
3. Right metering valve fails, jams, and prevents steering system movement (Item Z5)
4. Pilot's rudder-pedal linkage comes apart (Item X5)
5. Coordination unit fails, disconnecting the steering cables (Item Z6)
6. Cable terminal link breaks (Item X13)

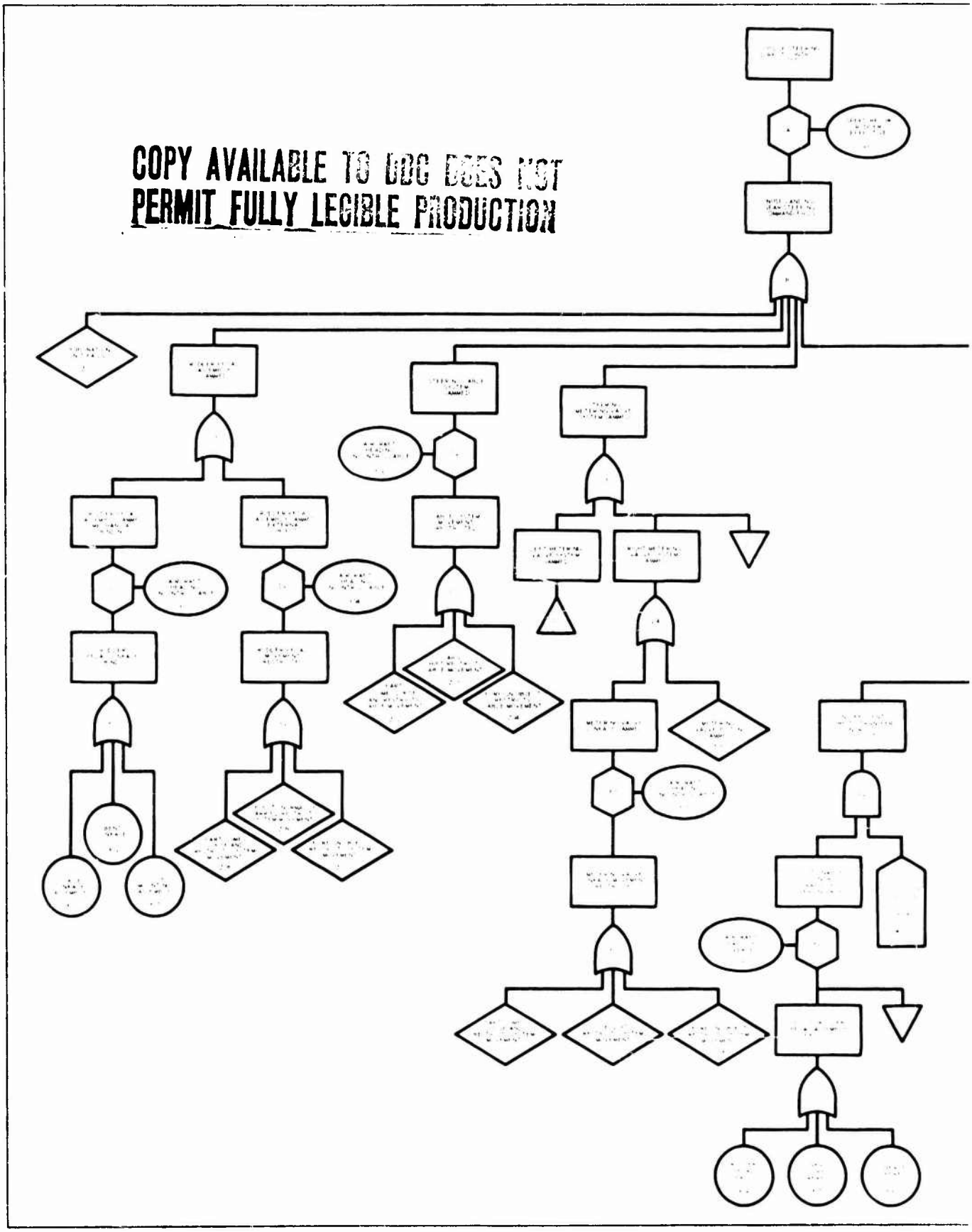
Each critical fault path was studied and simulated fixes were developed to alleviate the problems. For this example, the following fixes were assumed:

1. A protective cover was installed over the linkage (MTBF 10,000 hours).
2. A mechanism was installed that precludes one valve from failing and jamming the entire steering system.
3. Pilot's rudder-pedal linkage was redesigned with an improved locknut and bolt assembly (MTBF 77,000 hours).
4. Coordination unit was redesigned with more reliable components (MTBF 800,000 hours).
5. More reliable cable terminal links were installed (MTBF 400,000 hours).

The fault tree was modified with the new fixes (as input data), and the probability of occurrence of the undesired event was again determined (based on 100 million flight-hours). The statistical summary, Fig. 9, shows that the probability of occurrence of the undesired event has been reduced to 0.0043/100,000 hours.

Analysis of these data indicates that, based on 100 million hours of flight operation, the fixes were satisfactory. To improve the system further, additional hours could be simulated and the new critical fault paths identified. In actual practice, it is highly doubtful whether further improvements would be attempted. When a significant cost is involved in gaining an improvement, a cost-effectivness study would normally be made to determine the acceptability of the change.

The preceding example, although representing only a very small part of an aircraft and its mission, illustrates the technique that would be applied to the complete system and its operation.



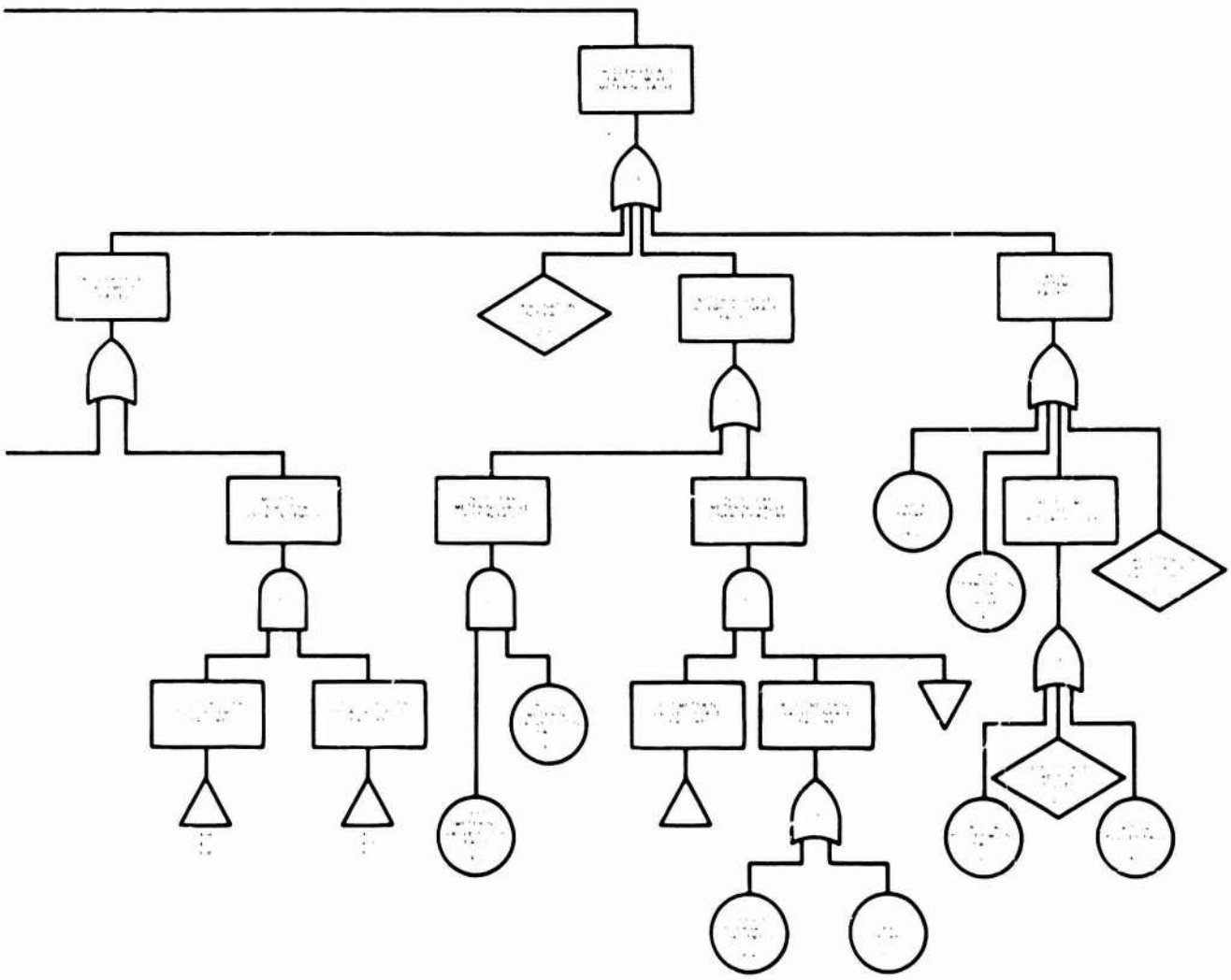


Figure 6. Example Fault Tree

NO.	COMPONENT	FAILURE MODE	MTBF HOURS	EFFECTIVE FAULT DURATION
X1	Rudder Pedal Linkage	Loose Linkage	$.10 \times 10^5$	100
X2	Rudder Pedal Linkage	Bent Link	$.30 \times 10^5$	100
X3	Rudder Pedal Linkage	Mount Loose	$.48 \times 10^5$	100
X4	Pilot's Rudder Pedal	Disconnects	$.24 \times 10^6$	5
X5	Pilot's Rudder Pedal Linkage	Comes Apart	$.77 \times 10^4$	5
X6	Rodder Pedal Link - Pilot's System	Link Breaks	$.20 \times 10^6$	5
X7	Copilot's Rudder Pedal	Disconnects	$.24 \times 10^6$	5
X8	Copilot's Rudder Pedal Linkage	Comes Apart	$.77 \times 10^4$	5
X9	Rudder Pedal Link - Pilot's System	Link Breaks	$.50 \times 10^5$	5
X10	Left Metering Valve Piston	Breaks	$.52 \times 10^3$	5
X11	Right Metering Valve Piston	Breaks	$.52 \times 10^3$	5
X12	Steering Cables	Breaks	$.50 \times 10^6$	5
X13	Cable Terminal Links	Breaks	$.80 \times 10^5$	5
X14	Right Metering Valve Linkage	Internal Failore	$.11 \times 10^4$	5
X14'	Left Metering Valve Linkage	Internal Failore	$.11 \times 10^4$	5
X15	Right Metering Valve Linkage	Link Breaks	$.20 \times 10^6$	5
X15'	Left Metering Valve Linkage	Link Breaks	$.20 \times 10^6$	5
X16	Critical Pulley Mount	Slackens Cable	$.10 \times 10^7$	5
X17	Critical Pulley	Slackens Cable	$.50 \times 10^6$	5
Z1	Coordination Unit	Jammed	$.25 \times 10^6$	5
Z2	Parts In Vicinity of Steering Cable System	Restricts Steering Cable System Movement	$.10 \times 10^6$	50
Z3	Cargo Shift	Restricts Steering Cable System Movement	$.10 \times 10^6$	5
Z4	Foreign Objects	Restricts Steering Cable System Movement	$.10 \times 10^6$	100
Z5	Right Metering Valve	Jammed	$.11 \times 10^6$	5
Z5'	Left Metering Valve	Jammed	$.11 \times 10^6$	5
Z6	Coordination Unit	Loosens or Disconnects Steering Cables	$.15 \times 10^6$	5
Z7	Cable Tension Regulator	Disconnects Cable Ron or Slackens Cable	$.37 \times 10^6$	5
Z8	Parts in Vicinity of Rudder Pedal Linkage	Restricts Rudder Pedal Linkage Movement	$.10 \times 10^4$	50
Z9	Objects Normally Carried	Restricts Rudder Pedal Linkage Movement	$.50 \times 10^5$	25
Z10	Foreign Objects	Restricts Rudder Pedal Linkage Movement	$.10 \times 10^5$	100
Z11	Foreign Objects	Forces Cable off Pulley	$.10 \times 10^7$	5
Z12	Parts in Vicinity of Right Metering Valve Linkage	Restricts Metering Valve Linkage Movement	$.10 \times 10^5$	10
Z12'	Parts in Vicinity of Left Metering Valve Linkage	Restricts Metering Valve Linkage Movement	$.10 \times 10^5$	10
Z13	Ice Dirt	Restricts Metering Valve Linkage Movement	$.10 \times 10^5$	10
Z13'	Ice Dirt	Restricts Metering Valve Linkage Movement	$.10 \times 10^5$	10
Z14	Foreign Objects	Restricts Metering Valve Linkage Movement	$.10 \times 10^5$	10
Z14'	Foreign Objects	Restricts Metering Valve Linkage Movement	$.10 \times 10^5$	10

OCCURRENCE		CONDITION	PROBABILITY
Y1	Velocity Below Rudder Effectiveness	Takeoff	.6
Y2	Pilot Unable to Control Aircraft Heading	Cable System Movement Restricted	.001
Y3	Pilot Unable to Control Aircraft Heading	Rudder Pedal Linkage Binding	.01
Y4	Pilot Unable to Control Aircraft Heading	Rudder Pedal Linkage Movement Restricted	.01
Y5	Pilot Unable to Control Aircraft Heading	Left Metering Valve Linkage Restricted	.01
Y5'	Pilot Unable to Control Aircraft Heading	Right Metering Valve Linkage Restricted	.01
Y6	Aircraft Rapidly Diverges	Pilot's Rudder Pedal Assemble Fails Disconnected	1

Figure 7. Fault Tree Input Data

ITEM NO.	NUMBER OF OCCURRENCES IN IMPORTANCE SAMPLE	FREQUENCY OF OCCURRENCE PER 100,000,000 FLIGHT HR.
X1	4	0.15
X2		
X3		
X4		
X5	14	2.0
X6		
X7		
X8		
X9		
X10		
X11		
X12	5	0.3
X13	13	1.9
X14		
X14'		
X15		
X15'		
X16		
X17	5	0.3
Z1	8	0.6
Z2		
Z3		
Z4		
Z5	12	1.5
Z5'	12	1.5
Z6	10	1.0
Z7	7	0.4
Z8	12	1.5
Z9		
Z10	4	0.15
Z11	4	0.15
Z12	4	0.15
Z12'	4	0.15
Z13	4	0.15
Z13'	4	0.15
Z14	4	0.15
Z14'	4	0.15

Undesired occurrences using Importance Sampling = 134
Equivalent occurrences after Importance Factor Conversion = 12.
No. of trials 1,000
Importance Factor 0.5
Flight time per trial – 100,000 hours
Probability of undesired event 0.012/100,000 hours
(Predicted on 100 million hours of simulated flight)

Item No. corresponds to events listed in Fig. 6

Figure 8. Statistical Summary of Fault Tree Simulation Results–Initial System Design

ITEM NO.	NUMBER OF OCCURRENCES IN IMPORTANCE SAMPLE	FREQUENCY OF OCCURRENCE PER 100,000,000 FLIGHT HR.
X1	4	0.15
X2	2	0.05
X3		
X4		
X5	4	0.2
X6		
X7		
X8		
X9		
X10		
X11		
X12	5	0.3
X13	8	0.6
X14		
X14'		
X15		
X15'		
X16	4	0.15
X17	5	0.3
Z1	8	0.6
X2		
Z3		
Z4		
Z5		
Z5'		
Z6	4	0.18
Z7	7	0.4
Z8	4	0.15
Z9	2	0.03
Z10	4	0.15
Z11	4	0.15
Z12	4	0.15
Z12'	4	0.15
Z13	4	0.15
Z13'	4	0.15
Z14	4	0.15
Z14'	4	0.15

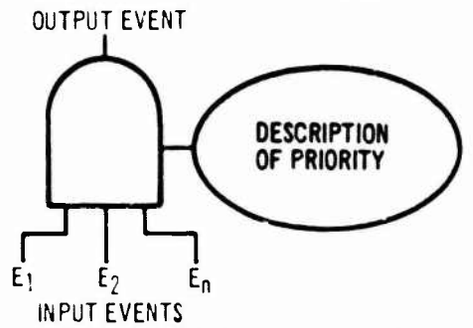
Undesired occurrences using Importance Sampling = 89
Equivalent occurrences after Importance Factor Conversion = 4
No. of trials 1,000
Importance Factor 0.5
Flight time per trial 100,000 hours
Probability of undesired event 0.0043/100,000 hours
(Predicted on 100 million hours of simulated flight)
Item No. corresponds to events listed in Fig. 6

Figure 9. Statistical Summary of Fault Tree Simulation Results–Revised System Design

SPECIAL LOGIC GATES

Previously, it was noted that in some situations the construction of a fault tree with only the basic logic gates can become prohibitively time consuming because of the number of variables that normally must be considered. In order to provide the analyst with the tools necessary to handle all types of situations, additional variations of basic logic gates have been developed.

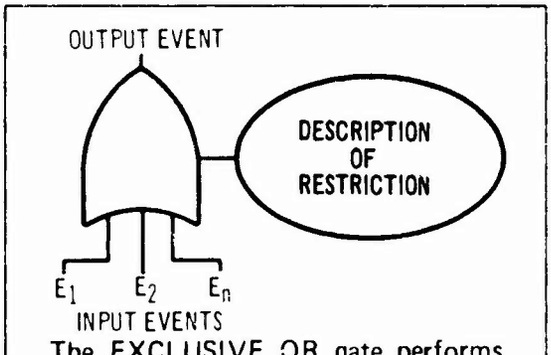
PRIORITY AND Gate.
This gate enables the analyst to consider events that must occur in a specified sequence, i.e., event E can occur only if A occurs before B. It is implicitly understood that if A occurs after B, event E does not occur. For example, consider an electrical system with provisions for automatically switching to an emergency power source when the primary source fails. Once the crossover has taken place, the automatic switch is no longer important and its subsequent failure will not result in loss of electrical power. Therefore, the only failures of the automatic switch that contribute to the undesired event are those that occur prior to failure of the primary source. This gate is described in Fig. 10.



The PRIORITY AND gate performs the same function as an AND gate with the additional stipulation that one event must precede the other.

Figure 10. Use of "PRIORITY AND" Gate

EXCLUSIVE OR Gate.
This gate describes the circumstance where event E can occur if A fails or B fails, but not if both fail. For example, assume that the analyst is concerned with the results induced by asymmetric thrust during a particular segment of a mission. The aircraft has two engines. Obviously, if both engines fail, the flight will be terminated abruptly, but for reasons other than asymmetric thrust. Therefore, a true measure would count "loss of engine No. 1" or "loss of engine No. 2," but not both. This gate is illustrated in Fig. 11.



The EXCLUSIVE OR gate performs the logical OR function but will not respond to the co-existence of two or more specified inputs.

Figure 11. Use of "EXCLUSIVE OR" Gate

VALVE, SUMMATION and DISCRIMINATOR GATES (These three gates are used in conjunction with one another.)

VALUE Gate.
The VALUE gate is used to reduce the number of logical OR gates required to simulate the cumulative effect of large numbers of contributing events. They describe the situation whereby event E occurs whenever the occurrence of A or B or C . . . or N cumulatively exceed a given value. Its purpose, therefore, is to serve as an analyst's shorthand symbol, permitting more rapid construction of fault trees.

It has no effect on the efficieny of calculation. For example, if the undesired event being analyzed is; "loss of pressurization due to leakage rate exceeding supply rate," there is an almost unlimited number of failure combinations that would result in this event. However, if each failure is assigned a value equivalent to leakage rate, and the values are summed as the failures occur, and then tested against the supply rate, the large number of combinations can be reduced to a workable number. The VALUE gate is shown in Fig. 12.

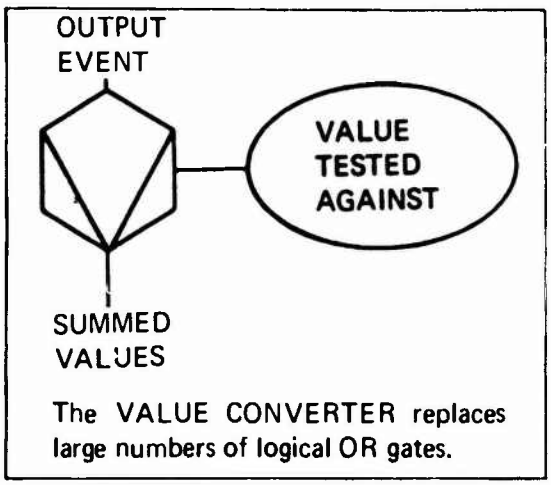


Figure 12. Use of "VALUE" Gate

SUMMATION Gate.
A SUMMATION gate is used to add the values from the individual VALUE gates. The SUMMATION gate is described in Fig. 13.

DISCRIMINATOR Gate.
The DISCRIMINATOR gate tests the value from the SUMMATION gate and allows the output event to occur if the input value is greater than or equal to the test value. It continually asks if the sum of the values from the VALUE gates, as summed by the SUMMATION gate, is greater than or equal to a preselected value N. If so, the event is permitted to occur. A DISCRIMINATOR gate must be used whenever a VALUE gate is used. the DISCRIMINATOR gate is shown in Fig. 14.

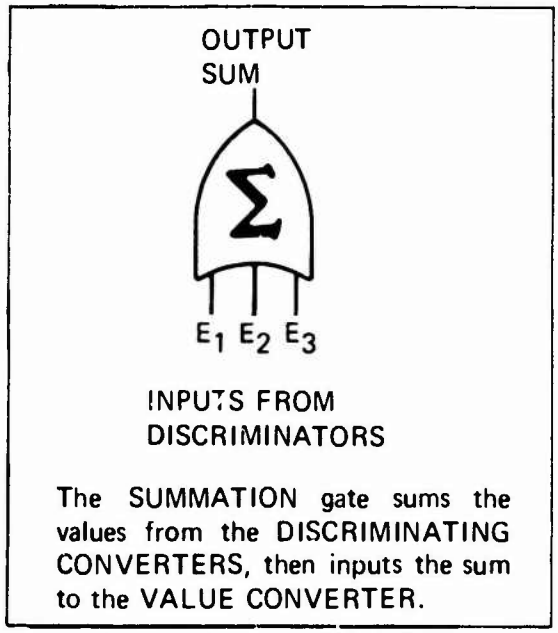


Figure 13. Use of "SUMMATION" Gate

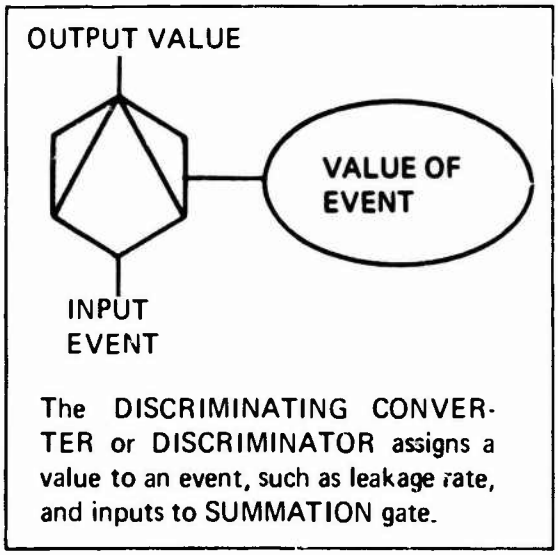


Figure 14. Use of "DISCRIMINATING CONVERTER"

An example of the application of the VALUE, SUMMATION, and DISCRIMINATOR gates to simplify the construction of a fault tree is shown in Fig. 15. The top half of the figure diagrams a simple problem without the use of these three gates. The bottom half diagrams the same problem using these gates.

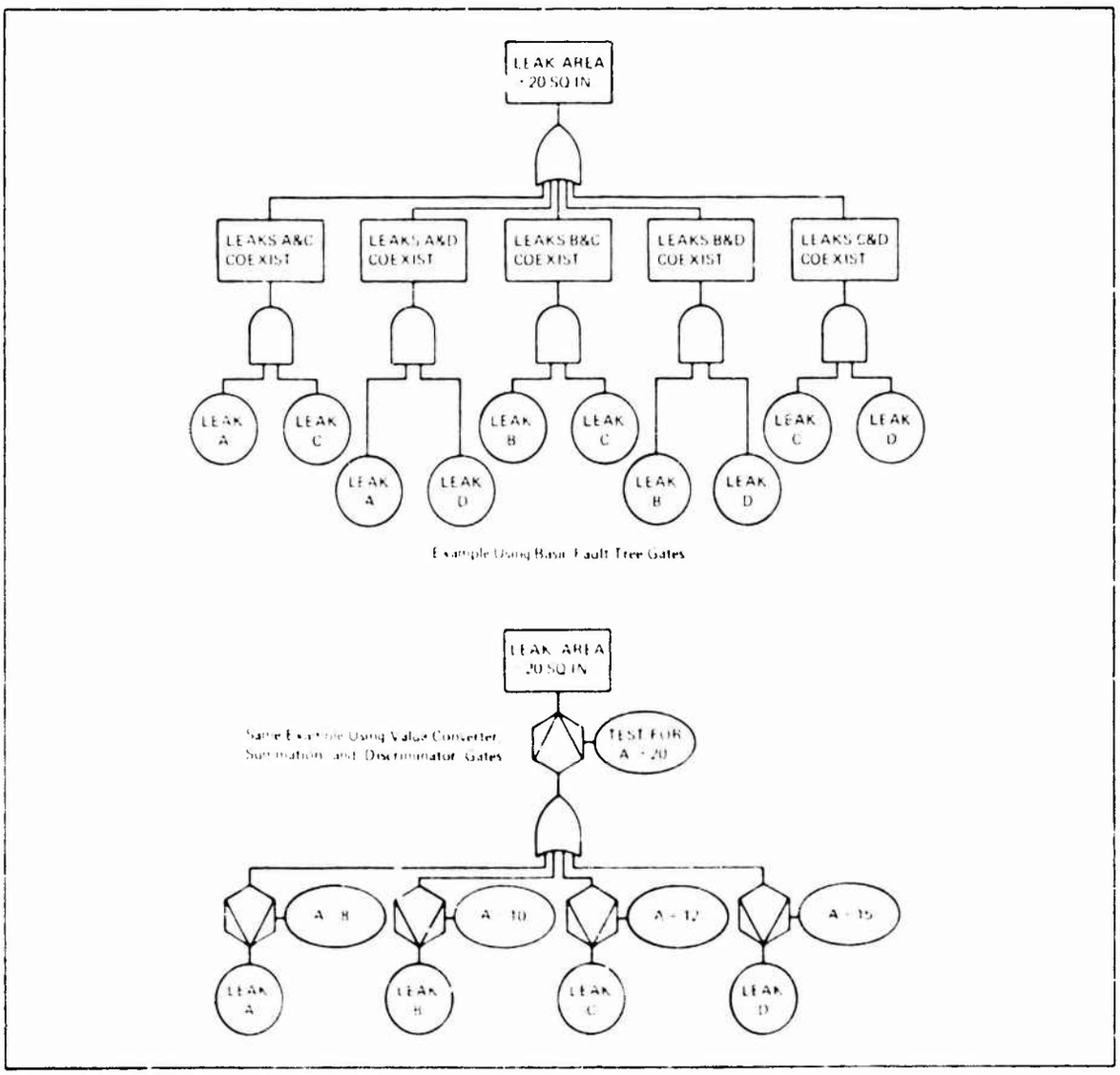


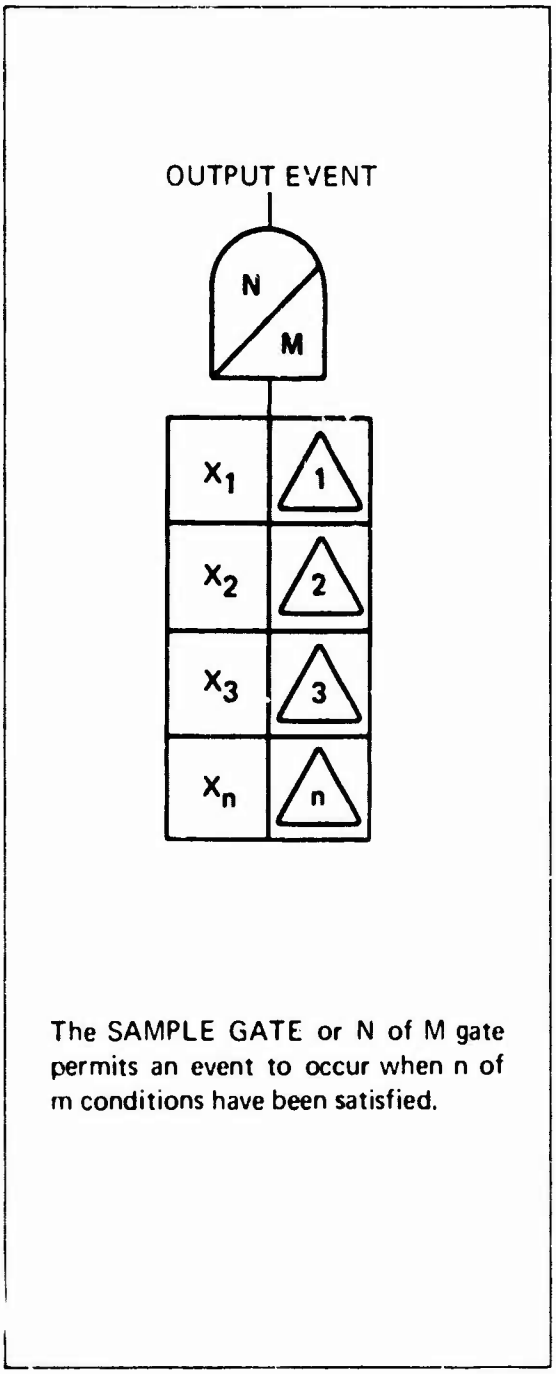
Figure 15. Example Application of "DISCRIMINATOR"

SAMPLE GATE.
The SAMPLE gate permits an event to occur when N of M conditions have been satisfied. This would otherwise require M!/N! (M-N)! inputs into an OR gate, each input consisting of N AND gates representing a different configuration of N of M conditions being satisfied. With the SAMPLE gate, all combinations are portrayed with one simple device.

For the special case where N = 1, we have a circumstance where any one of the M conditions will satisfy the requirements for event occurrence, or a simple M input OR gate. For the special case where N = M, we have a circumstance where all M conditions must occur to satisfy the criteria, or a simple M input AND gate. The SAMPLE gate is shown in Fig. 16.

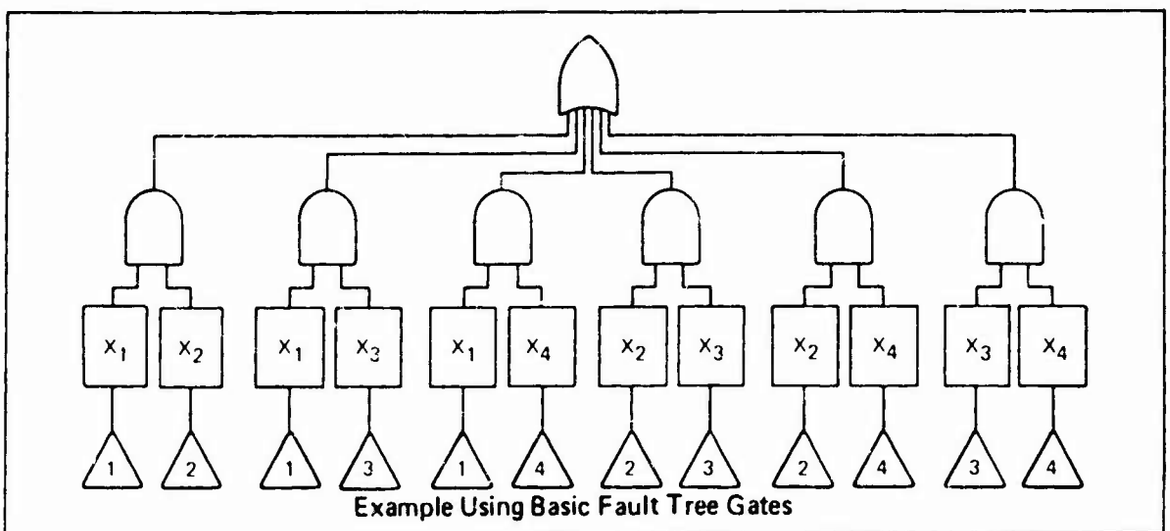
An example of the application of the SAMPLE gate to simplify the construction of a fault tree is portrayed in Fig. 17. The top half of the figure illustrates the equivalent of the SAMPLE gate portrayed conventionally. The bottom half depicts the same problem using this gate.

MATRIX GATE.
This gate is another simplifying drafting symbol. It is used when one wishes to portray AND gates inputting into INHIBIT gates which, in turn, input into an OR gate. That is, it simplifies portrayal of the condition where the output event may be caused N percent of the time that A and B occur or M percent of the time that A and C occur, or L percent of the time that B and C occur. For example, one can portray the event of losing proper trim when it is assumed the pilot will react properly 80 percent of the time that A and B occur, 95 percent of the time that A and C occur, and 90 percent of the time that B and C occur. The MATRIX gate is shown in Fig. 18. The numbers in the gate symbol identify a matrix of input values described elsewhere on the diagram.

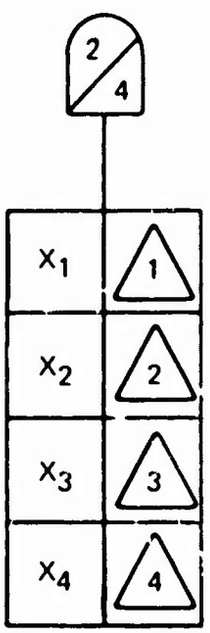


The SAMPLE GATE or N of M gate permits an event to occur when n of m conditions have been satisfied.

Figure 16. Use of "SAMPLE" Gate

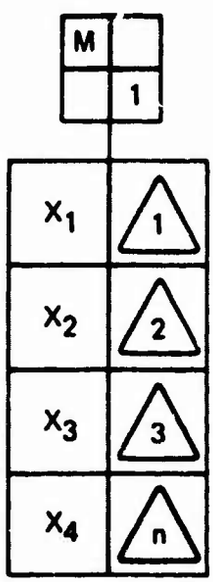


Example Using Basic Fault Tree Gates



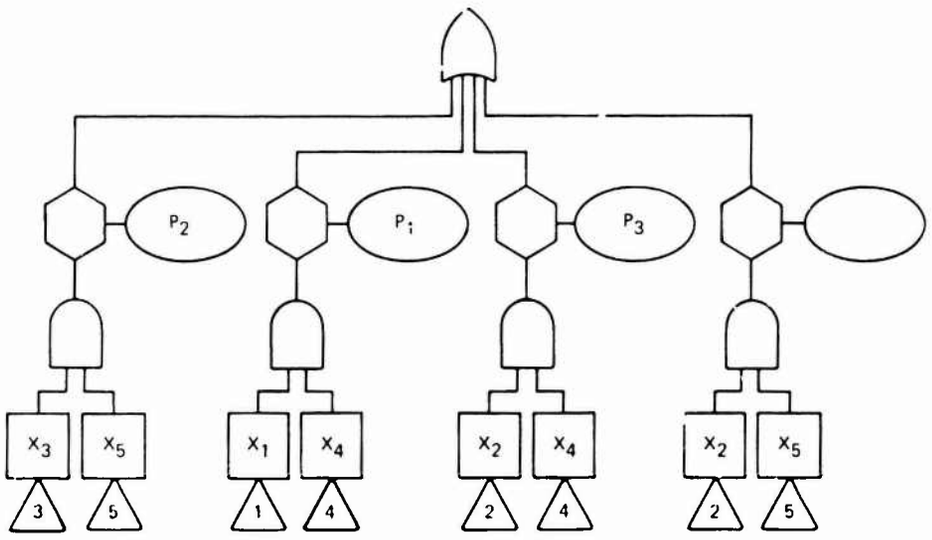
Same Example Using Sample Gate

Figure 17. Example Application of "SAMPLE" Gate

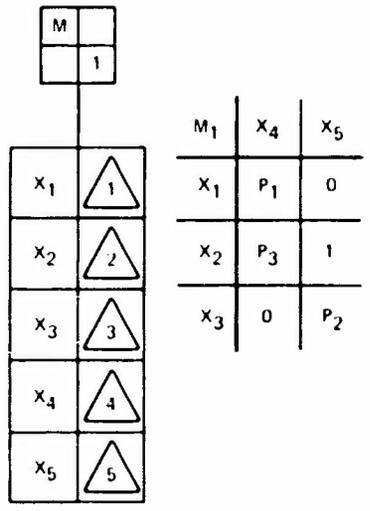


The MATRIX gate replaces series of AND gates inputting to an INHIBIT gate through an OR gate.

Figure 18. Use of "MATRIX" Gate



Example Using Basic Fault Tree Gates.



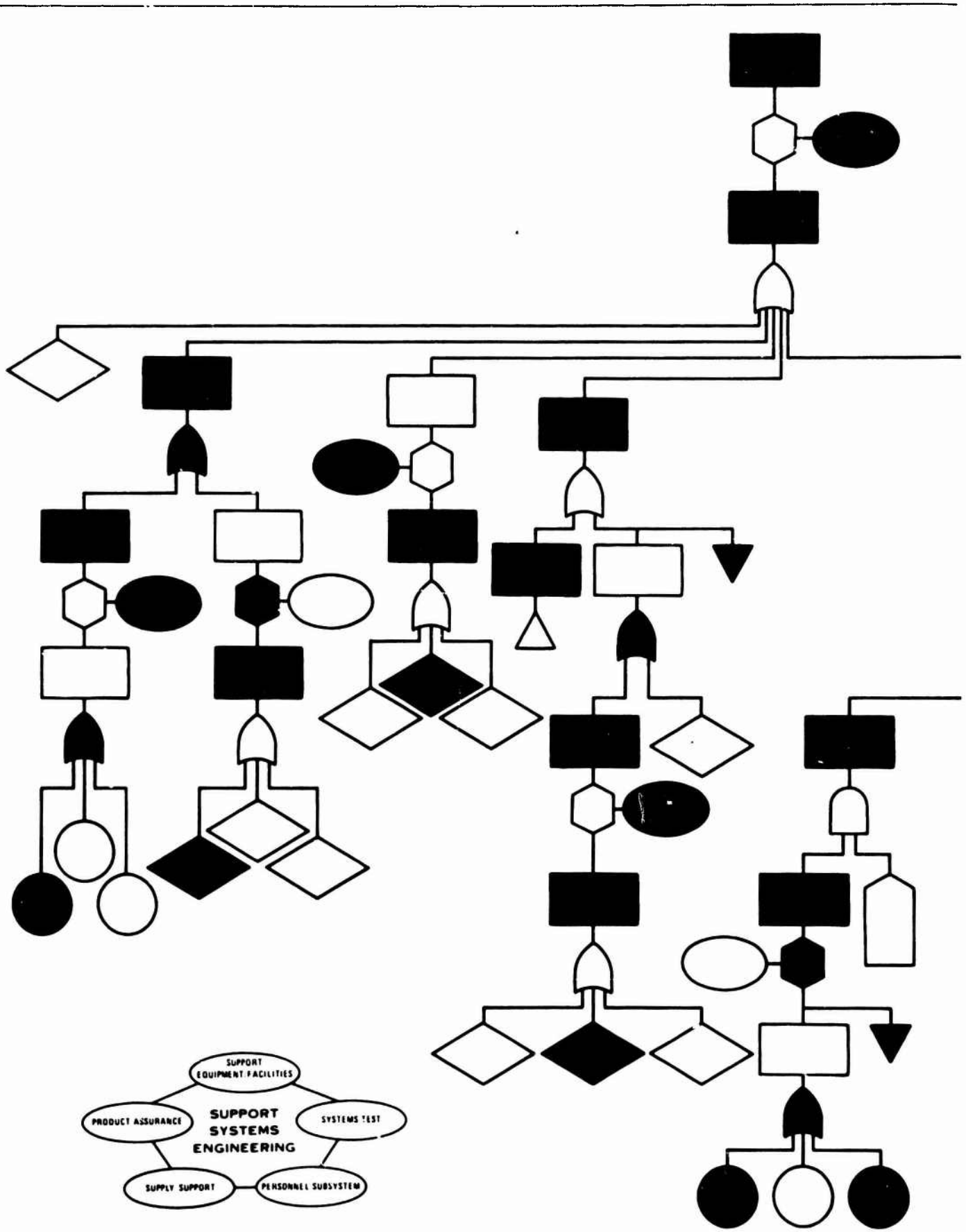
M_1	X_4	X_5
X_1	P_1	0
X_2	P_3	1
X_3	0	P_2

Same Example Using MATRIX Gate

An example of the application of the MATRIX gate is diagrammed in Fig. 19. The upper portion of the figure illustrates the equivalent of a MATRIX gate portrayed conventionally. The lower half shows the same problem using the MATRIX gate.

Figure 19. Example Application of "MATRIX" Gate

If further information is desired on the technique described herein, or related matters, please contact H. D. Trettin, Boeing Aerospace Company, P. O. Box 3999, Seattle, Washington 98124, Tel. 206-773-1270.


SUPPORT EQUIPMENT FACILITIES
PRODUCT ASSURANCE
SUPPORT SYSTEMS ENGINEERING
SYSTEMS TEST
SUPPLY SUPPORT
PERSONNEL SUBSYSTEM