FEATURE ARTICLE

George Novacek

Designing for Reliability, Maintainability, and Safety

Part 1: Getting Started

If you think that designing for reliability and maintainability are just steps in the development process that eat up more budget without adding any value, then you might want to listen up because George has evidence that proves otherwise.

Electronic equipment designers are familiar with the concepts of reliability, maintainability, and safety (R/M). But sadly, we use these disciplines much less than they deserve, especially in consumer product development. Many people find them boring and think they chew up a big chunk of the precious development budget but add no value.

In this article, I want to show you the basic steps in performing R/M analyses and how they do add value to your design. You'll learn to appreciate their benefits and, hopefully, come to the conclusion that your time is well invested using them. Not all manufacturers spend the time to ensure good design, but as you will learn, we could be buying better, safer products at lower prices if they did.

To be effective, the analyses must be performed concurrently and as an integral part of product development. I will take you through development of a hypothetical controller to illustrate the fundamentals of the R/M engineering approach. I set up the project to demonstrate the use of R/M tools, not the design of a controller. So, some design decisions are tailored to exaggerate R/M aspects. For the same reason, some issues have been simplified or omitted.

HAZARD ANALYSIS

Let's assume you are contracted to develop an electronic controller for a hot tub, to maintain its water temperature at 100°F (±2°). It is heated by a gas-fired burner capable of raising the water temperature to the boiling point quickly. The hot tub supplier (your customer) should start the project by performing a system hazard analysis. And, you should know enough about the system to prepare your own. The requirements pertinent to the subsystem become a part of the performance specification.

The simple analysis shown in Table 1 has only two potential failures. The failures and their permitted probabilities of occurrence per hour of operation are the result of a trade-off between requirements and cost.

With the possibility of personal injury at hand, I'm not aware of any system where it would be considered an acceptable risk to allow such a design weakness. Performing the hazard analysis and implementing its results reduces risk and shows reasonable care and prudent design in court (just in case).

The probability of 10^{-9} failures per hour of operation generally is accepted as "never." With the exponential fault distribution, which is most popular in electronics and yields a constant failure rate, it represents a 50% chance of experiencing such a failure after about 79,000 years of continuous operation.

The second line of Table 1 shows noncritical subsystem failures when the hot tub is not as hot anymore. You don't want such failures but can live with them. Statistically, all components will fail at some point. As you will learn during this design exercise, decreasing the probability of a failure is expensive. When there is no potential for personal injury, the decision boils down to the manufacturing cost versus potential customer unhappiness, cost of service, warranty, maintainability, life cycle cost, and so on.

Imagine that the controller fails. It doesn't matter why, now you have a customer complaint and must send out a maintenance technician. If it happens during the warranty period, you pay for the repair. With the probability of failure pegged at 10^{-5}/h as a design goal, you must expect a failure every 100,000 h of the controller operation. But, suppose sales take off and during the next few years you sell 10,000 units. If they run 24 h per day, you can expect one failure every 10 h.

Is it possible to lower the failure rate by an order of magnitude? One million hours of mean time between failures (MTBF) would drastically reduce the service cost but will be expensive to achieve. What is the cost of improving MTBF compared to the savings in maintenance and warranty cost? Will there be enough parts to service the equipment several years from now, recognizing that the current life cycle of microelectronic parts is merely five years or less?

Let's say you must extend a five-year warranty. Based on the probable failure rate, you can estimate the warranty cost. How will it affect profitability? R/M analyses also help make these business decisions. Provided they are performed concurrently with the design, their results are implemented in a closed-loop system for optimal results. Discovering the laws of statistics after the product introduction to the market may be revealing, but usually too late.

RELIABILITY FUNDAMENTALS

To fully appreciate the aspects of reliability, you need to review the fundamentals of failure prediction. Because electronic components can be most often modeled by constant failure rate (λ), which is the characteristic property of exponential failure distribution, you won't have to go into the gory details of statistical analysis. The mathematics will be straightforward.

Suppose a sample population of a component you are interested in is tested while you record observed failures, plotting them against time of their occurrence. You can plot the number of occurrences within given, short, time intervals to obtain a frequency distribution plot. Or, you can record the cumulative number of failure occurrences against the time as you proceed with the test (cumulative frequency distribution).

The cumulative frequency distribution of the majority of electronic components will be exponential and resembles the curves in Figure 1. The mathematical model for the frequency distribution is called probability density function (PDF) and for exponential distribution is expressed as $f(t) = \lambda \times e^{-\lambda t}$. The cumulative frequency of distribution is modeled by the cumulative distribution function (CDF):

$$F(t) = \int_0^t f(y)dy$$
$$0 \le t \le \infty$$

where $f(y)dy$ is a dummy variable of integration. For the exponential distribution, you can write the following functions for PDF and CDF, respectively:

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

$$F(t) = 1 - e^{-\lambda t}$$

where λ is a single unknown that defines the fault distribution. You can calculate reliability (the number of surviving units) as:

$$R(t) = 1 - F(t)$$

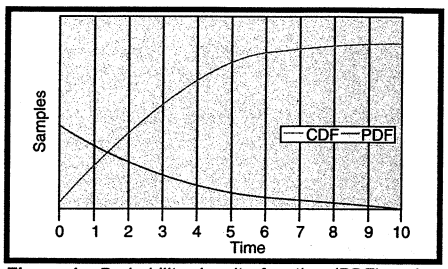


Figure 1—*Probability density function (PDF) and cumulative distribution function (CDF) are the results of exponential failure distribution in components and are characteristic of the constant failure rate typical of electronic parts.*

Then, use this to arrive at the expression for an instantaneous failure rate:

$$h(t) = \frac{f(t)}{R(t)}$$

and solve for constant λ:

$$h(t) = \frac{\lambda \times e^{-\lambda t}}{e^{-\lambda t}} = \lambda$$

A component following an exponential life distribution exhibits the same probability of failure in the next hour regardless of whether it is new or used. It does not age or degrade with use. Failure occurs at a constant rate, unrelated to the hours of use.

This important, seemingly illogical concept allows you to gain equivalent information from testing 10 units for 10,000 h as if from 1,000 units for 100 h. It also means that the "impossible" 10^{-9} failure is as likely to happen in the first 5 min. of operation as 114,000 years from now.

If, based on observation, you suspect that the failure rate does depend on the time used, it may be because of wear caused by improper derating or the exponential fault distribution does not apply for this part.

A measure of reliability more commonly used at the user level for irreparable parts is mean time to failure (MTTF). For components with exponential life distribution, this is a reciprocal of λ. For assemblies where a failed component can be replaced, MTBF is appropriate. For exponential fault, distribution also equals $1/\lambda$.

Figure 2 shows the well-known bathtub diagram. The diagram consists of three

Failure description	Failure effect	Maximum probability
The controller fails to turn off the burner when the water reaches 102°F.	A critical failure must not happen under any circumstances, personal injury may result.	$<10^{-9}$
The controller fails to turn on the burner when the water temperature drops below 98°F.	During a noncritical failure, the tub is not useable and the system is no longer available. Customer dissatisfaction results.	$<10^{-5}$

Table 1—*Even a simple hazard analysis matrix is an effective tool for identifying system weaknesses and potential hazards that the design must eliminate.*

Reliability Models For Electronic Components

The calculated value is λ_p, which represents the predicted number of failures per 10^6 hours. The reliability model for microelectronic circuits (ICs) states that:

$$\lambda_P = (C_1 \times \pi_T + C_2 \times \pi_E) \times \pi_Q \times \pi_L$$

where:

C_1 = die complexity; 0.14 for the PIC controller and 0.020 for the regulator 7805.

π_T = temperature coefficient. Assuming the junction temperature $T_j < 100°C$ for both ICs, it will be 1.5 for the PIC controller and 16 for the regulator.

C_2 = a constant based on the number of pins. 0.0034 is used for the PIC with 8 pins and 0.0012 for the 3-pin regulator.

π_E = environmental constant. Assume the equipment will operate in a "ground fixed" environment, a benign location with average ambient temperature of 25°C, not exceeding 45°C.

π_L = learning factor; 1 for ICs more than two years in production.

π_Q = quality factor. This is the most controversial coefficient. For military screened components it is between 1 and 2, but climbs to 10 for commercial components. Many critics have established that the penalty for commercial, off-the-shelf parts is unrealistically high, especially when taking into account modern manufacturing processes.

For diodes, the equation looks like:

$$\lambda_P = \lambda_b \times \pi_T \times \pi_T \times \pi_S \times \pi_C \times \pi_Q \times \pi_E$$

where:

λ_b = base failure probability related to the construction; 0.0012 for switching and general-purpose diodes, 0.0030 for power rectifiers, and 0.0013 for transzorbs.

π_T = temperature coefficient; 3.9 for junction temperature $T_j <$ 70°C.

π_S = is based on stress 1.0 for transzorbs and 0.054 for other diodes in the system, provided they are not exposed to more than 30% of their rated characteristics.

π_C = contact construction factor; 1.

π_Q = 8.0 for plastic encapsulated devices.

π_E = environmental constant; 6.0 for the "ground fixed" environment.

Next is the solenoid driver power MOSFET (less than 1-W dissipation):

$$\lambda_P = \lambda_b \times \pi_T \times \pi_A \times \pi_Q \times \pi_E$$

where:

λ_b = base failure probability related to the construction; 0.012 for power MOSFET.

π_T = temperature coefficient; 2.3 for junction temperature $T_j <$ 70°C.

π_A = 1.5 for switching applications.

π_Q = 8.0 for plastic encapsulated devices.

π_E = environmental constant; 6.0 for the "ground fixed" environment.

Resistors' reliability calculates as follows:

$$\lambda_P = \lambda_b \times \pi_T \times \pi_P \times \pi_S \times \pi_Q \times \pi_E$$

where:

λ_b = base failure probability related to the construction; 0.0037 for RLR resistors and 0.0019 for thermistors.

π_T = temperature coefficient; 1.3 for resistor operation less than 50°C and 1 for the thermistor.

π_P = is determined by the dissipated power; 0.44 for 100 mW and greater, 1 for 1 W, and 0.44 for the thermistor.

π_S = 1.1 for stress factor 0.4 (i.e., you don't operate the device at more than 40% of its rated characteristics). It equals 1.0 for the thermistor.

π_Q = 3.0 for devices without established reliability.

π_E = environmental constant; 6.0 for the "ground fixed" environment.

Capacitors' reliability is expressed as:

$$\lambda_P = \lambda_b \times \pi_T \times \pi_C \times \pi_V \times \pi_{SR} \times \pi_Q \times \pi_E$$

where:

λ_b = base failure probability related to the construction; 0.00051 for fixed, metallic film capacitors and 0.00040 for tantalum capacitors.

π_T = temperature coefficient; 1.6 for capacitor operation less than 50°C

π_C = a factor for capacitance; 0.81 for the fixed capacitors assumed to be 0.1 µF and 1.6 for the electrolytic capacitors assumed to be 10 µF.

π_V = 1 for stress factor 0.3.

π_{SR} = 1 for both types.

π_Q = 3.0 for devices without established reliability.

π_E = environmental constant; 10.0 for the "ground fixed" environment.

The next device on the list is the transformer:

$$\lambda_P = \lambda_b \times \pi_T \times \pi_Q \times \pi_E$$

where:

λ_b = base failure probability; 0.022 for low-power transformers.

π_T = temperature coefficient; 1.4 for operation less than 50°C.

π_Q = 3.0 for devices without established reliability.

π_E = environmental constant; 6.0 for the "ground fixed" environment.

And finally, for quartz crystals:

$$\lambda_P = \lambda_b \times \pi_Q \times \pi_E$$

where:

λ_b = base failure probability related to the crystal frequency; 0.022 for 10 MHz.

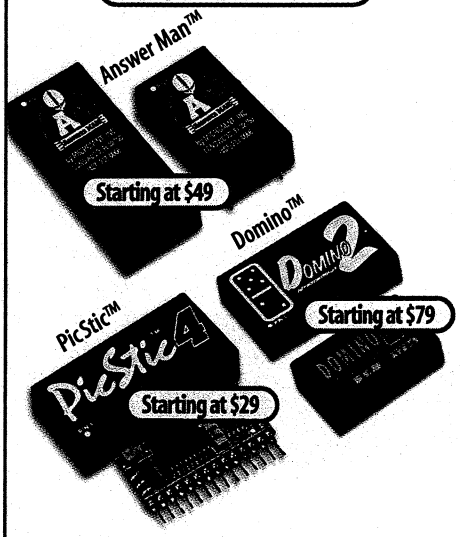
π_Q = 2.1 for nonmilitary devices.

π_E = environmental constant; 3.0 for the "ground fixed" environment.

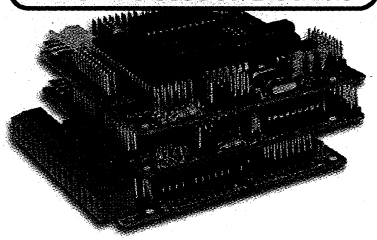
The results of the calculations are tabulated in Table 2.

Tools for the Imagination

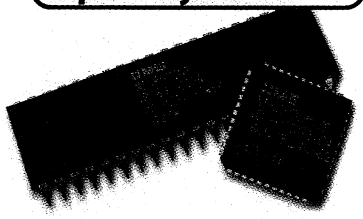
Micro Modules



RTC Processor Boards



Specialty Products



With dozens of embedded controllers and countless configurations, we can turn your imagination into reality.

For a complete look at our product line, visit our website or call for a free catalog at

(800) 635-3355.

Micromint

www.micromint.com

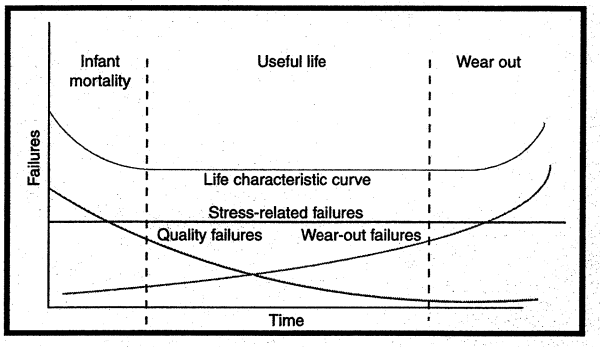


Figure 2—*The product life cycle characteristic curve is composed from the constant failure rate of components, increased at the beginning by infant mortality and at the end by wear. (This figure is not to scale.)*

curves and three distinct areas. The quality curve is predominant during the initial life period and is often referred to as infant mortality, which is the result of design, handling, or workmanship problems.

Infant mortality effects can be reduced by using robust designs and manufacturing process control. At the end of the manufacturing process, a good burn-in or environmental stress screening (ESS) period will weed out the majority of the failures. Products shipped from the factory should be past the infant mortality curve.

The useful life period is characterized by stress-related failures, in other words, the MTTF or MTBF. The infant mortality curve can bottom out within a few days of "shake 'n bake," but the useful life period is counted in years.

Finally, at the end of their life, components begin to fail because of wear. Although common in mechanical systems, well-derated electronic systems seldom reach this period.

RELIABILITY PREDICTION

Reliability prediction analysis results in definition of λ_p, the predicted failure rate, which is expressed as a number of failures per 10^6. This number forms the basis for other R/M analyses. In most electronic systems with a constant failure rate, the MTBF and MTTF are the reciprocals of λ_p stated in hours. If a device is not repairable (e.g., a failed transistor), MTTF is used. If it can be repaired, MTBF is used.

Because they attempt to forecast the future, reliability prediction methods have been the subject of many heated debates. Several schools of thought exist, each having adherents and just as many opponents. This article is not the forum for getting into the thick of the debate. I'll discuss MIL-HDBK-217F because it's a widely recognized model based on the constant failure rate. [1]

Remember that you are extrapolating historical or test data into the future. This data is used as a yardstick for performance evaluation and improvement. So, you need to understand what the numbers mean and must not look at them as the only objective.

Tweaking the numbers will be self-defeating. Field returns are always a more powerful statement of performance than statistical predictions. Both you and the customer need to understand that the failure probability of 10^{-6} is not a guarantee of a million-hour, failure-free operation. A failure can occur in the first or one-millionth hour. On the positive side, reliability models are conservative, the equipment outperforms the statistics.

Let's start with preliminary design of the controller. You'll calculate its predicted reliability, put it through FMECA and FTA analyses, and review the results (see Figure 3).

For this example, I used a CMOS PIC micro operating from a 5-V power supply provided by a three-terminal 7805 regulator. The water temperature is detected by thermistor R3 and the gas-fired water heater is controlled by solenoid valve (SV) L1. The valve contains an internal freewheeling diode to suppress back-EMF kick and is switched off and on by power MOSFET Q1.

Diodes D2 and D4 clamp the analog input of the PIC within 0 to 5 V to protect the micro and prevent ESD damage. One of the built-in ADCs reads the thermistor voltage. The second ADC monitors the current through the solenoid valve to provide short-circuit protection and to monitor operation. The valve is external and supplied by the system integrator, so exclude it from the reliability calculation; although, it is an important player for safety. Tranzorb D5 protects the MOSFET driver against voltage

Component	Description	$I_p/10^6$ hours	MTTF
R1	Small resistor RLR	2.7936E-02	3.5795E+07
R2	Small resistor RLR	1.0794E-02	9.2647E+07
R3	Small resistor RLR	1.0032E-02	9.9681E+07
R4	Small resistor RLR	2.7936E-02	3.5795E+07
R5	Small resistor RLR	1.0794E-02	9.2647E+07
R6	Current sensing resistor	6.3492E-02	1.5750E+07
C1	Tantalum capacitor 10 mF	3.0720E-02	3.2552E+07
C2	Tantalum capacitor 10 mF	3.0720E-02	3.2552E+07
C3	Metallic capacitor 0.1 mF	1.9829E-02	5.0432E+07
C4	Metallic capacitor 0.1 mF	1.9829E-02	5.0432E+07
C5	Metallic capacitor 0.1 mF	1.9829E-02	5.0432E+07
C6	Metallic capacitor 0.1 mF	1.9829E-02	5.0432E+07
C7	Metallic capacitor 0.1 mF	1.9829E-02	5.0432E+07
C8	Metallic capacitor 0.1 mF	1.9829E-02	5.0432E+07
Q1	Power MOS-FET P_D = 1 W	1.9872E+00	5.0322E+05
U1	7805 regulator	1.1680E-01	8.5616E+06
U2	PIC12C672 microcontroller	1.8160E-01	5.5066E+06
D1	1A bridge rectifier	1.2131E-02	8.2436E+07
D2	Signal diode (1N914)	1.2131E-03	8.2436E+08
D3	Signal diode (1N914)	1.2131E-03	8.2436E+08
D4	Signal diode (1N914)	1.2131E-03	8.2436E+08
D5	Transzorb	2.4336E-01	4.1091E+06
X1	Quartz crystal	1.3860E-01	7.2150E+06
T1	Transformer 120 V primary	5.5440E-01	1.8038E+06
Controller total		3.5691	280,180 h

Table 2—*Preliminary failure rate calculation also indicates that you are on the right track and that the customer's specification is achievable.*

transients, which could be the result of the freewheeling diode failure.

To calculate the predicted failure rate, you could use one of several expensive programs. Some methods even extract components and their operating conditions out of the schematic capture program, simplifying the chore that generations of engineers have performed manually. My program is a small circuit and performing the calculation by hand will be good exercise. As stated previously, you're using the MIL-HDBK-217F model with exponential distribution, assuming a constant failure rate. All the values and calculations come from that source. Read the "Reliability Models for Electronic Components" sidebar for the details.

Immediately it is apparent that the resulting failure rate and MTBF of nearly 300,000 h satisfy the 10^{-5} system availability requirement the customer defined in the hazard analysis. Can it be improved? Let's take a closer look, because this would minimize future warranty claim costs, maintenance requirements, and improve customer satisfaction.

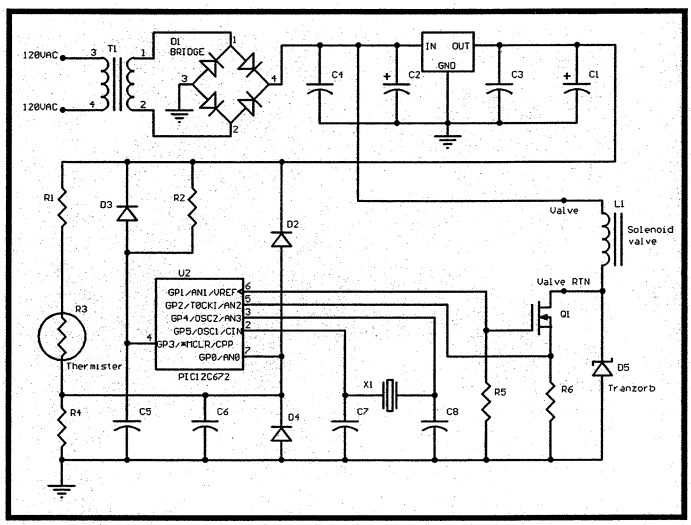


Figure 3—*Here's my first run at the controller schematic diagram. The design is simple, straightforward, and appears to do what is needed.*

All components are well derated (i.e., working at less than 30% to 40% of their specified ratings). Further derating will have a minimal effect on their reliability improvement. But, five components have failure rates that are greater than the rest: Q1, U1, U2, D5, and T1. What can you do?

Q1, U1, and U2 work at conservatively estimated junction temperature T_j = 100°C. Keeping the ambient operating temperature at 27°C and with efficient heat sinking, T_j can be reduced to 50°C. Heat is the reliability killer and even a small reduction will have a significant effect.

Transzorb D5 and diodes D2 and D3 conduct current during infrequent transients only. You can reduce their contribution by applying a duty cycle. Design T1 to run at a lower temperature and you'll improve its reliability.

Implementation of these steps will increase the MTBF to 714,000 h (λ_p = 1.4). And, for the remaining analyses, you will use the results of these calculations to evaluate product safety.

George Novacek has 30 years of experience in circuit design and embedded controllers. He currently is the general manager of Messier-Dowty Electronics, a division of Messier-Dowty International, the world's largest manufacturer of landing-gear systems. You may reach him at gvovacek@nexicom.net.

SOFTWARE

Reliability calculations are available on the *Circuit Cellar* web site.

REFERENCES

[1] U.S. Department of Defense, *Reliability Prediction of Electronic Equipment*, MIL-HDBK-217F, General Policy Series, no. 14T.

S.E.R. subcommittee, *Automotive Electronics Reliability Handbook*, Society of Automotive Engineers, Warrendale, PA, 1987.

P. Tobias and D. Trindale, *Applied Reliability*, Van Nostrand Reinhold, NY, NY, 1986.

U.S. Department of Defense, *Electronic Reliability Design Handbook*, MIL-HDBK-338, General Policy Series, no. 542.

U.S. Department of Defense, *Maintainability Program for Systems and Equipment*, MIL-HDBK-47OB, Washington D.C.: Government Printing Office, 1995.

FEATURE ARTICLE

George Novacek

Designing for Reliability, Maintainability, and Safety

Part 2: Digging Deeper

Having covered the consequences of not making your design safe and reliable, George is ready to get up to his neck in the details of the hot tub controller application. Relax, turn up the jets, and get ready to toast the success of your next design.

In Part 1 of this series, I talked you through designing a simple hot tub controller. You calculated its predicted reliability and discovered that it satisfies the MTBF design criteria. Reliability was improved by moving the controller to where the ambient temperature excursions combine with components' heat sinking, resulting in lower junction temperature than originally estimated.

Now that you have a controller that performs the desired function, it's time to satisfy the safety requirements. This is not as easy as it seems. I've stated many times that achieving the product's desired functionality is a fraction of the design effort. More effort is expended to make the design safe. So, let's discuss the details.

BEING PREPARED

Failure mode and effects analysis (FMEA) is a bottom-up review of a system. In this analysis, you examine components for their failure modes, notice how the failures propagate through the system, and study their effects on the system's behavior. This leads to design review and possibly changes to eliminate weaknesses.

By adding the criticality column in the FMEA work sheet, the analysis becomes FMECA (failure modes, effects, and criticality analysis). In most systems, it is not necessary to examine every component. You can rearrange the design into functional blocks and, when needed, consider individual component failures within functional blocks that may be critical. Take a look at Figure 1. This is the circuit of Figure 3 of Part 1 broken into four functional blocks, A, B, C, and D.

The work sheet shown in Table 1 is a standard format that engineers often tailor to fit their specific requirements. This matrix is simplified, limited only to issues you need to consider. The first column identifies the failure. For a more complicated system, you would have a separate database of the failures with reference pointers to the work sheet. The letter identifies the functional block, the number, and the individual failure of the block.

The next three columns are self-explanatory. The method of detection includes built-in test capability and status reporting. Your simple, hypothetical controller has some, but as I'll explain, every fault must be detected, therefore the design needs to be modified accordingly.

There are only two criticality levels, high and low. High criticality failure causes the heater to stay on to heat the water above 102°F; a noncritical failure causes loss of heating, and consequently, the use of the system is lost.

The probability column will assign a probability number to the fault taken from the reliability prediction in Table 2. To accomplish that, simply identify the components in the functional block, add their respective λ_p, and multiply by 10^6.

Observation is the only detection method of malfunction. This isn't acceptable for critical failure, when the water temperature exceeds the maximum limit and must be provided by the built-in test (BIT) function.

What do the FMECA results show you? They indicate that satisfying the 10^{-5} system availability will not be a

problem. The reliability prediction has already shown that. But, the FMECA brought several important facts concerning the design to the surface.

One fact is that failure A2 needs to be watched carefully (don't change the design until you finish full analysis). Failure of the power supply, just a cold joint of the grounding pin of U1, will likely damage the controller and could cause critical water overheating.

A3 means the power supply puts out less VDC than expected. It could be a half wave rectified AC. You have no idea how the controller will react to this. You could perform more analyses, going from block to component level, analyzing failure modes and effects of every component, and then try to improve the reliability of the components potentially responsible for critical failures. However, as the probability number shows, you are almost three orders of magnitude away from satisfying the critical performance (10^{-9} is required for water overheating). Therefore, a more drastic measure, other than beefing up components' specs, is needed.

B1 and B2 show that there is a two orders of magnitude deficit in satisfying the critical requirement. The microcontroller isn't the problem. Software is a potential culprit. Assume the software has been properly verified and validated and its reliability is not an issue. But, even 100% correct software can go on a tangent because of external effects. Therefore, the software probability of failure is pegged at $<10^{-10}$, which is normal.

Defects in the temperature sensor, block C, must be detected by the microcontroller running a plausibility check on the values. Two checks can be performed here: the value must be within a plausible range and the rate of change must not be greater than expected from the system. Your system will be fail passive, meaning that if the microcontroller detects invalid data, heating will shut down. The mechanical design must make sure thermistor R3 is exposed to the water temperature at all times.

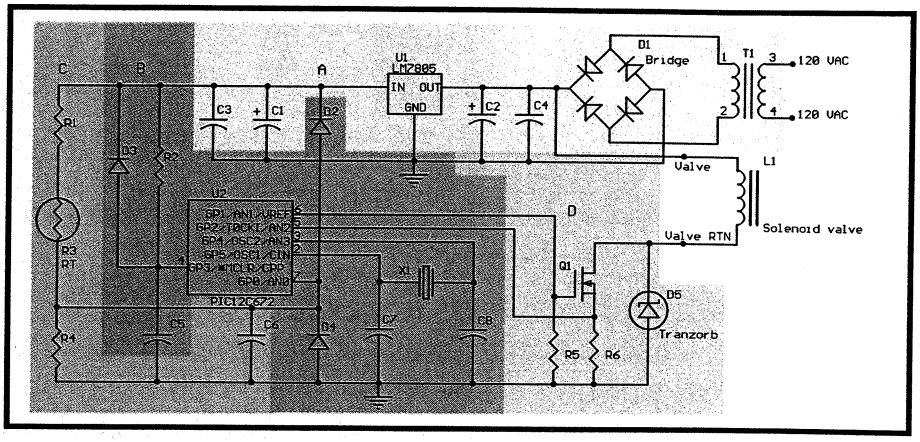


Figure 1—*To perform FMECA, the diagram is divided into functional blocks. It is a bottom-up review of the design. Consider functional failures and examine how they propagate to the system level. Generally, functional blocks give sufficient detail, but check out individual components only if there is a critical failure.*

I won't dwell on nonelectrical issues. Other than the mechanical influence, there's no defined failure mode where a thermistor value would remain electrically correct but fail to modify its resistance according to its temperature.

Block D is monitored for the solenoid valve (SV) current through R6. This allows detection and protection from short and open circuits. However, Q1 is a critical component. If it fails by shorting SV to ground, a critical fault will result. A similar situation exists for transzorb D5, SV, and SV's wiring (more about this later). D5 is not stressed unless there is a transient, and therefore, its effect can be adjusted by a duty cycle.

I'll give you one last tip. It's advantageous to have an indicator to announce the controller failure. Moving on, for the last step of the design evaluation, you'll perform a fault tree analysis (FTA).

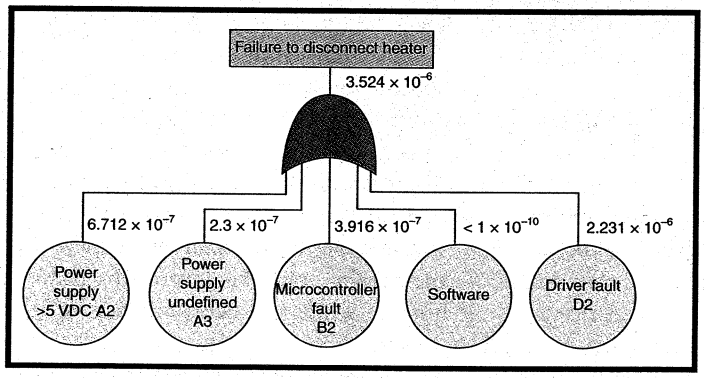


Figure 2—*The fault tree analysis supplements FMECA. This is a top-down view of the system. You identify critical failures and consider which causes will contribute to them.*

FAULT TREE ANALYSIS

In many respects, the FTA and FMECA could be used interchangeably, because they are different representations of the same data. The difference is that the FMECA is a bottom-up and the FTA is a top-down graphical analysis. The FTA starts with the top event you're interested in, then builds the fault tree using Boolean logic and symbols. By adding known failure probabilities, the same used when creating the FMECA, you arrive at the probability of the event of interest. As with the FMECA, the analysis can be performed on the functional block as well as at the component level. Using Boolean logic, probabilities fed into an OR gate will be mathematically added, while the ones fed into an AND gate will be multiplied:

$$P_{OR} = P_1 + P_2 \ldots + P_N$$
$$P_{AND} = P_1 \times P_2 \ldots \times P_N$$

The top event you are interested in is the uncontrolled heating of the water. Because there is only an OR gate in the FTA, any one event in the circle can cause the top event. Having calculated the failure rate for the uncontrolled heating as $\lambda = 3.524 \times 10^{-6}$, you can calculate the probability of this failure occurring:

$$P_F = 1 - e^{-\lambda \times t}$$

System: hot tub controller				Document number			Revision
Function: water temperature control				Environment: ground fixed			Date
Operation phase: all				Prepared			Checked
Failure no.	Failure mode	Possible cause	Failure effects	Method of detection	Criticality	Probability λ/h	Remarks
A1	Output = 0 V	Can be caused by a failure of any component within functional block A or an external short	Loss of water heating	Observation	Low	7.844×10^{-7}	
A2	Output > 5 V	Failure of T1 or U1	Potential damage to U2, unpredictable effects. Maybe loss of or continuous heating	Observation	High	6.712×10^{-7}	
A3	Output out of tolerance	C1, C2, C3, C4, D1, U1	High ripple or out-of-spec operating voltage; unpredictable.	Observation	High	2.3×10^{-7}	Can be eliminated by monitoring the power supply health and forcing reset if outside limits.
B1	Output continuously 0	U2, C5, C7, C8, R2, D3, software	Loss of water heating	Observation	Low	3.9×10^{-7}	
B2	Output continuously 1	U2, C5, C7, C8, R2, D3, software	Continuous heating	Observation	High	3.9×10^{-7}	This means the Microcontroller block is not working. Its output could be stuck in either state.
C1	Temperature sensing not working	R1, R3, R4; Any device open or short circuit	Loss of water heating	Input signal plausibility check by microcontroller observation	Low	2.296×10^{-7}	Resistor network is designed such that a short or open of any device takes the signal out of plausible range.
C2	Temperature sensing not working	Thermal link between water and R3 lost	Continuous heating	Observation	High	Undefined	Mechanical design issue
D1	No SV drive	Q1, R5, R6	Loss of water heating	Microcontroller monitors Q1current; observation	Low	2.304×0^{-6}	
D1	Continuous SV drive	Q1, D5	Continuous heating	Microcontroller monitors Q1current; observation	High	2.231×10^{-6}	Can be detected but not remedied by the system

Table 1—*The analysis data is organized in the FMECA work sheet, which makes it easy to review assumptions and conclusions.*

For t = 10 years, that's 87,600 hours of operation.

$$P_F = 1 - e^{-3.5424 \times 10^{-6} \times 10 \times 365 \times 24} = 0.266$$

Or, for P_F = 0.5 (50% chance of uncontrolled heating), it takes 22 years of operation. But that's not good enough for a system that can potentially cause injury. Using the equation above, calculate $\lambda = 1 \times 10^{-9}$, which is for the specification requirement. This would give even odds for the uncommanded heating after 79,000 years.

WHAT'S NEXT?

For the uncommanded heating, you are nearly three orders of magnitude removed from the specification goal of $\lambda = 10^{-9}$. It's unrealistic to come close to this goal by improving the components' reliability. But, what if you could feed the top event in Figure 2 into an AND gate? ANDing it with another signal of merely 2.8×10^{-4} probability of failure would do the trick (see Figure 3).

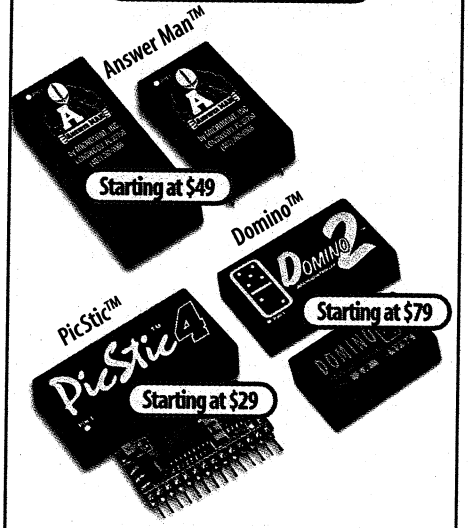
This is how high safety and reliability is achieved in systems by redundancy. You have to sacrifice the overall MTBF as you add components, but critical functions will perform better. The simplest approach, it might seem, would be to add a mechanical thermostat in series with Q1 to open the circuit at 102°F. However, every fault that could cause a critical failure must be either prevented from happening or detected. Adding a function that may or may not be available does not solve the problem.

The thermostat in the SV path doesn't solve the problem. Its failure can't be detected, meaning it has a dormant failure. As long as the electronic controller works properly, the thermostat could be defective yet you would never know. Conversely, the thermostat could be controlling the hot tub while the electronic controller is dead.

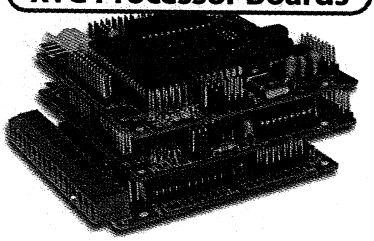
The most common solution is to double the processing channels and revert to a safe state, in this case the

Tools for the Imagination

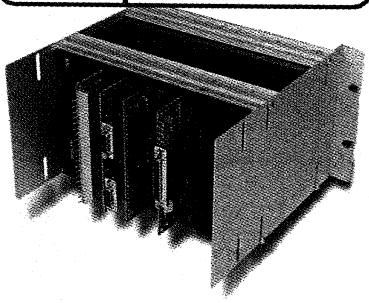
Micro Modules



RTC Processor Boards



BCC Expansion Boards



With dozens of embedded controllers and countless configurations, we can turn your imagination into reality.

For a complete look at our product line, vist our website or call for a free catalog at **(800) 635-3355.**

Micromint

www.micromint.com

Component	Description	$\lambda_p/10^6$ h	MTTF
R1	Resistor	1.0794×10^{-2}	9.2647×10^{7}
R2	Resistor	1.0794×10^{-2}	9.2647×10^{7}
R3	Thermistor	3.8760×10^{-3}	2.5800×10^{8}
R4	Resistor	1.0794×10^{-2}	9.2647×10^{7}
R5	Resistor	1.0794×10^{-2}	9.2647×10^{7}
R6	Resistor	6.3492×10^{-2}	1.5750×10^{7}
R7	Resistor	1.0794×10^{-2}	9.2647×10^{7}
R8	Resistor	1.0794×10^{-2}	9.2647×10^{7}
R9	Resistor	1.0794×10^{-2}	9.2647×10^{7}
R10	Resistor	1.0794×10^{-2}	9.2647×10^{7}
R11	Resistor	1.0794×10^{-2}	9.2647×10^{7}
R12	Resistor	1.0794×10^{-2}	9.2647×10^{7}
R13	Resistor	1.0794×10^{-2}	9.2647×10^{7}
R14	Resistor	1.0794×10^{-2}	9.2647×10^{7}
R15	Resistor	1.0794×10^{-2}	9.2647×10^{7}
R16	Resistor	1.0794×10^{-2}	9.2647×10^{7}
R17	Resistor	1.0794×10^{-2}	9.2647×10^{7}
C1	Electrolytic capacitor	3.0720×10^{-2}	3.2552×10^{7}
C2	Electrolytic capacitor	3.0720×10^{-2}	3.2552×10^{7}
C3	Solid capacitor	1.9829×10^{-2}	5.0432×10^{7}
C4	Solid capacitor	1.9829×10^{-2}	5.0432×10^{7}
C5	Solid capacitor	1.9829×10^{-2}	5.0432×10^{7}
C6	Solid capacitor	1.9829×10^{-2}	5.0432×10^{7}
Q1	MOS-FET	4.4352×10^{-1}	2.2547×10^{7}
Q2	MOS-FET	4.4352×10^{-1}	2.2547×10^{7}
U1	Regulator	1.9000×10^{-1}	5.2632×10^{7}
U2	Micro	9.4800×10^{-2}	1.0549×10^{7}
U3	Comparator	5.3200×10^{-2}	1.8797×10^{7}
U4	Reset IC	9.4000×10^{-3}	1.0638×10^{8}
D1	Bridge rectifier	9.2192×10^{-3}	1.0847×10^{8}
D2	Signal diode	1.3001×10^{-7}	7.6914×10^{12}
D3	Transzorb	8.2368×10^{-6}	1.2141×10^{11}
D4	Signal diode	1.3001×10^{-7}	7.6914×10^{12}
D5	Transzorb	8.2368×10^{-6}	1.2141×10^{11}
D6	Signal diode	1.3001×10^{-3}	7.6914×10^{8}
D7	Signal diode	1.3001×10^{-3}	7.6914×10^{8}
D8	Transzorb	8.2368×10^{-6}	1.2141×10^{11}
T1	Transformer	2.7720×10^{-1}	3.6075×10^{6}
X1	Crystal	1.3860×10^{-1}	7.2150×10^{6}
F1	Fuse	2.0000×10^{-2}	5.0000×10^{7}
Controller total		1.8611×10^{0}	537,320 h

Table 2—*The final failure rate calculation proves the reliability expectations will be met.*

heater disconnect, if the two channels disagree. Because you have no way of knowing which channel is correct, you can't continue operating. But if a fail operative system is needed, at least three processing channels with a majority vote will do the job.

When designing a redundant system, it is often advantageous (sometimes required) to design the channels differently to avoid common mode failures in channels. Similarly, you must avoid having a single point of failure, for example, feeding all channels from the same power supply where >5-V output could cause damage to the channels or uncommanded heating. Figure 4 is the simplified diagram of the hypothetical controller, now improved so that it meets the safety requirements.

Several circuit modifications were made to satisfy the specification. Modifications included adding transzorb D3 (5 V) and fuse F1 to the power supply. If the power supply output exceeds 5 V, the transzorb will conduct and the excessive current will blow the fuse.

The simple RC reset network was replaced with a Motorola MC34064 low-voltage sensor/reset IC. It will hold the PIC controller in reset any time the supply voltage drops below the TTL level.

To recover one microcontroller I/O pin, an external clock oscillator is used. GP2 and GP4 were switched to make the internal counter available for the monitor. And, a second SV driver, Q2, was added for a totem pole driver

topology. A hardware monitor that uses a single quad comparator, such as LM139, was added too.

How does the circuit work? The PIC controller reads the thermistor output and by driving Q1, turns on and off the SV to maintain set temperature. It also performs a sanity check on the thermistor input. A short or open fault of any component within the thermistor bridge would cause the output voltage to move out of the plausible range. Similarly, an abrupt change in temperature, inconsistent with the rating of the heater and water mass, would indicate a fault condition.

Parallel with the microcontroller, the sensor voltage is fed into comparators A, B, and C of U3, forming the front end of the monitor circuit. Thermistor R3 with R1 and R4 represent a single point failure. But, because that failure is detectable by both the processor and monitor, a single sensor will satisfy the safety needs. Resistors R4 and R17 isolate a fault in either the processor or monitor to stop it from propagating to the other channel.

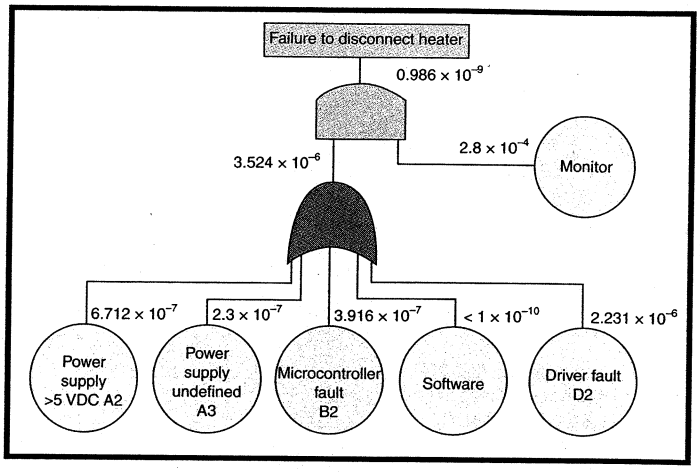


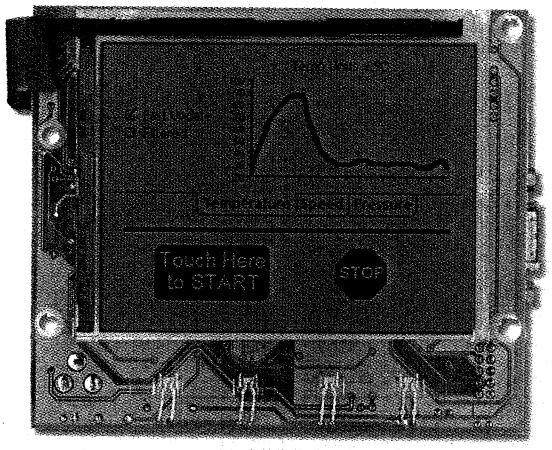
Figure 3—*FTA shows that by adding a monitor to the heater controller, the top-level event now requires two failures to happen simultaneously. The probability of such an occurrence has decreased significantly.*

All four comparators' outputs are ORed; LM139 has open collector outputs and is ideal for this purpose. When the temperature exceeds the maximum limit of 102°F, comparator A turns off Q2, thus removing power from the SV in case the microcontroller fails. Similarly, voltage comparators B and C form a window for plausibility testing of the temperature sensor. If it goes outside the predetermined limits, Q2 will be turned off regardless of the microcontroller action.

Now comes the difficult part. As I said, there must be no dormant failure in the system. All faults must be detected (it assumes only one fault happens at a time and that you're starting with a fully functional unit).

How do you make sure the comparators work properly and that Q2 can disconnect the SV? While heating, the microcontroller injects short pulses through diode D6 into the comparators. The voltage levels need to be adjusted accordingly through a resistor

Now... GUI and LCD Control in a Single Package!



The Easy GUI™ Starter kit (STK-GT320) also includes our µHTML™ Compiler, sample HTML files, and sample images. Plus, the onboard flash is factory programmed with µHTML pages so you can be up and running – right out of the box!

- ***1/4 VGA, 3.8-inch, Monochrome Display*** – with ultra-bright backlight and fully-integrated analog touch panel
- ***Dedicated GUI Controller*** – manages the GUI, interacts with the user, and controls the LCD
- ***Processor Independent*** – easily interfaces to most microcontrollers (8/16/32-bit and even DSPs)
- ***HTML-Based GUI*** – converts from HTML, JPEG, and GIF into small, quickly-executable Amulet µHTML™ pages
- ***Replaces Traditional GUI Library*** – No library porting, complex GUI programming, or RTOS required
- ***Standard RS232 Interface*** – Up to 115.2 Kbps, cable included
- ***64K-Bytes of Onboard Flash Memory*** – For storing hundreds of Amulet µHTML pages that you create
- ***Partitioned Design*** – for parallel development, quick design changes, easy testing, and product migration

Easy GUI™ Starter Kit (STK-GT320) – Only $399!

Amulet Technologies
GUI Engines For Embedded Systems

©2001 Easy GUI and µHTML are Trademarks of Amulet Technologies. U.S. and Foreign Patents Pending.

AmuletTechnologies.com (408) 244-0363

System: hot tub controller				Document number			Revision
Function: water temperature control				Environment: ground fixed			Date
Operation phase: al				Prepared			Checked
Failure no.	Failure mode	Possible cause	Failure effects	Method of detection	Criticality	Probability λ/h	Remarks
A1	Output = 0 V	Can be caused by a failure of any component within functional block A or an external short	Loss of water heating	Observation	Low	4.26517×10^{-7}	
A2	Output > 5 V	Failure of T1 or U1 and D3	Potential damage to U2, unpredictable effects. Maybe loss of or continuous heating	Observation and BIT	High	2.9621×10^{-7}	The failure is detected by the monitor and the heater disconnected. A double failure is needed for this condition, but dormancy exists.
A3	Output out of tolerance	C1, C2, C3, C4, D1, U1	High ripple or out-of-spec operating voltage; unpredictable.	Observation and BIT	Low	4.1592×10^{-7}	The power supply health is monitored. Reset is forced if the voltage is outside limits.
B1	Output continuously 0	U2, U4, X1, software	Loss of water heating	Observation	Low	2.3340×10^{-7}	
B2	Output continuously 1	U2, U4, X1, software	Continuous heating	Observation and BIT	High	2.4280×10^{-7}	The microcontroller lock is monitored by hardware and its erratic operation results in heater disconnect.
C	Temperature sensing not working	R1, R3, R4; Any device open or circuit short mechanical disconnect N/A	Loss of water heating control	Input signal plausibility check by microcontroller observation	High	6.6879×10^{-7}	Resistor network is monitored by BIT. Mechanical disconnect of the thermistor is prevented by design.
D1	No SV drive	Q1, R5	Loss of water heating	Observation BIT	Low	4.5431×10^{-7}	
D2	SV continuously on	Q1 or both transzorbs D5 and D8 failed short	Continuous heating	Observation BIT	High	4.5431×10^{-7}	Failure of either transzorb detected by BIT
E	Continuous SV drive or no drive	U3, Q2, R6, R7, R9–R17, C5, D6, D7	Continuous heating or loss of water heating	BIT observation	High	6.9058×10^{-7}	Monitored by microcontroller.

Table 3—*The final FMECA work sheet shows the design is safe. Faults are detected and the system shuts down.*

will be detected as such. An open circuit failure remains inconsequential until the corresponding MOSFET is damaged by a transient, at which time the condition will be detected. There also could be a far-fetched failure of the microcontroller whereby it is stuck in a loop driving the SV continuously while periodically recharging C5.

As you see, even a simple design can quickly snowball into a major project when safety becomes an issue. In this case, you may be able to show that a microcontroller failure with these symptoms is highly improbable, or you can take steps to detect such a condition. A timing window comparator is one way and a voltage comparator to track the two gate drive signals is another way of detection.

Although there is always room for safety improvement, you confront the law of diminishing returns quickly. Therefore, it's necessary to exercise good judgment and make sure you don't go overboard, increasing not only the product cost, but also complexity and occurrence of nuisance alarms. In more complex systems, you need to use tools such as testability analysis to achieve necessary fault coverage without going overboard. In simple, commercial systems such as this one, a lot can be accomplished by simply having an audible alarm to sound when system control is lost.

Scenix Tools

Now supporting 100mhz

SX-ISD-100

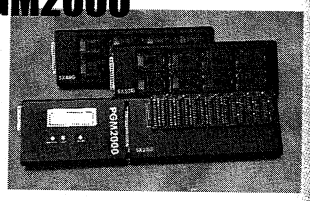
In-system Debugger for SX18/20/28/48/52
Source Level Debugging for SASM, SXC
Built-in programmer
Real-time Breakpoint
Conditional Animation Break
External Break Input and External Clock Input
Frequency Synthesizer, 25Khz to 120Mhz*
Selectable Internal Frequencies
Software Animation Trace
Parallel Port Interface
Runs under Win 95/98/2000/NT4
Comes with SASM Assembler
*SX-ISD-100 model, SX-ISD to synthesize to 75mhz but support external oscillator input to 90-95mhz

All tools are qualified and used in-house by Scenix Semiconductor

PGM-SX

Parallel Port Interface
40-pin socket
Program device in socket or in-circuit
Win 95/98/NT software
SASM assembler
Optional SOIC, SSOP and QFP programming sockets

PGM2000



Stand-alone 8-gang programmer
On-line operation via parallel port
Detachable 8-socket program adapters
DIP, SOIC, SOP and QFP adapters
Programming voltage adjustable in 0.1V
Codes and fuse reside securely in EEPROM of Master Control Unit
Comes with Win 95/98/NT software
From $900

*Advanced*TransdAtA
14330 Midway, Suite 128, Dallas, Texas
Tel 972.980.2667 Fax 972.980.2937
Email: atc1@ix.netcom.com

www.adv-transdata.com

divider network. This injects a fault into the monitor. At the same time, the microcontroller looks at the SV drive current as seen across R6. It must drop to zero for the duration of the test pulse. The microcontroller does the same, driving Q1 directly to verify it can turn off the SV. Because the mechanical parts of solenoid valves have 30- to 60-ms reaction time, this test pulse has no effect on the heater. If the microcontroller discovers the system response is not as expected, it will shut down the system.

Now that you know the monitor works, how do you know the microcontroller works, too? Comparator D does the job for you. Through D7, capacitor C5 is being continuously recharged every time the fault pulse is injected into the monitor, similar to a watchdog timer. It discharges through resistor R14, and if it's not recharged in time because of a fault in the microcontroller circuit, the comparator disables Q2.

But how do you prove the circuit is working? Every few seconds during the heating cycle, the microcontroller allows C5 to discharge. At this point, it must detect a drop in SV current across R6. But, what if the microcontroller is stuck high, keeping C5 charged? Then the test pulse into devices A, B, and C will stay high and Q2 will be off.

Close examination of the circuit shows that there still are several potential dormant failures. For example, transzorb D3 protecting the voltage regulator and D5 across the SV driver. To monitor D3, you may include a power-up diagnostic procedure to inject fault into the system. Careful circuit analysis may reveal that the transzorb is insufficient for the overvoltage protection and that a crowbar circuit would be more appropriate. Either way, you may consider detecting the power supply failure by a different method.

Because the analog comparators can handle 30 V_{CC}, they can be designed to detect the power supply as well as the microcontroller failure. The fuse is a different story—there is no nondestructive way to test it. You'll have to settle for the crowbar (or a transzorb) to handle the overcurrent indefinitely, or to blow a PCB track, or cause some other acceptable damage.

The potential D5 failure can be corrected by using transzorbs D5 and D8, as shown in Figure 5. A short circuit failure of either one will have the same effect as Q1 or Q2 failure and

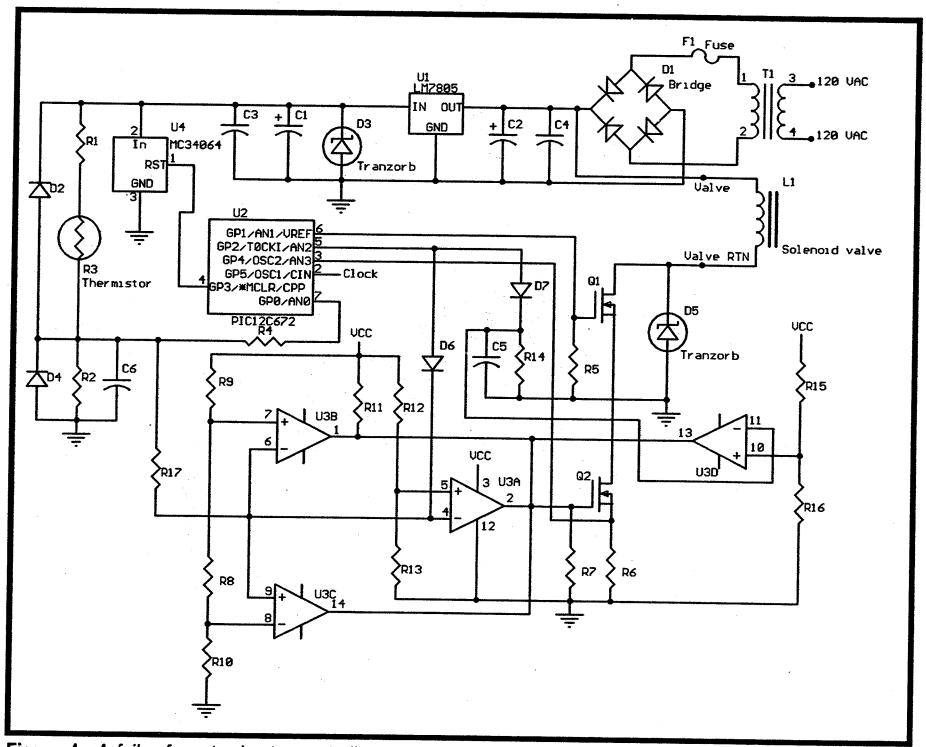


Figure 4—*A fail-safe water heater controller requires additional monitoring of circuits. This is my first attempt. It still does not satisfy the requirements.*

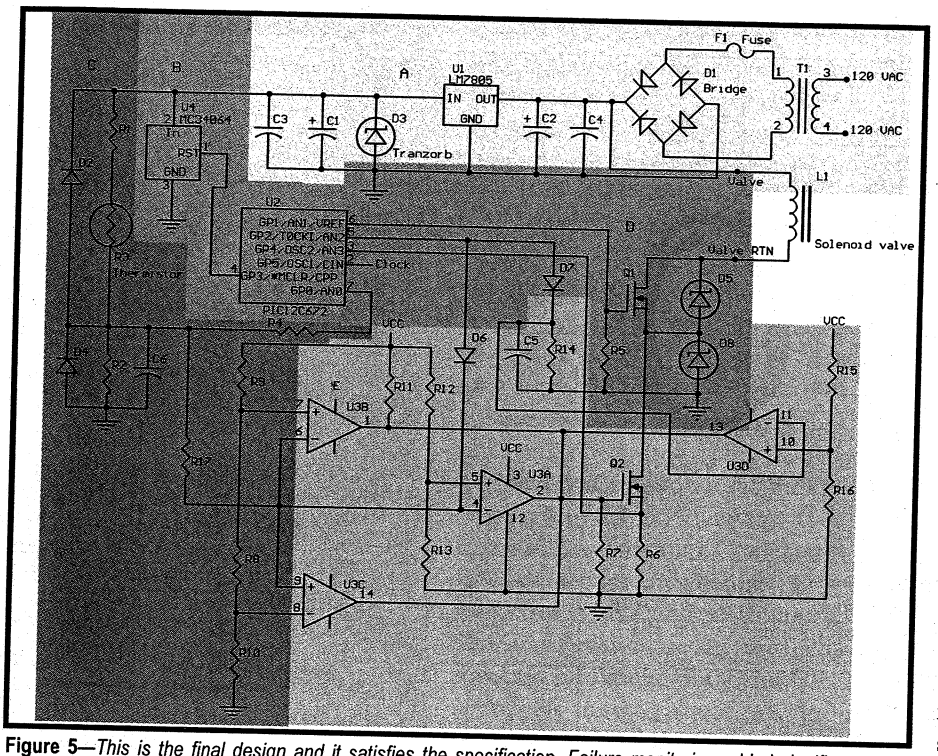


Figure 5—*This is the final design and it satisfies the specification. Failure monitoring added significant complexity to the original design.*

It's time for a word about watchdog timers, often touted as the guarantor of microcontrollers' faultless performance. They are useful, but have limitations and by themselves do not guarantee product safety. The watchdogs integral within the microcontroller are no more reliable than the micro. Although they may be useful to restart the program if it skips the rail because of a software bug or external transient, if there is a bona fide fault on the substrate, watchdogs are most likely toast.

External watchdog timers such as Maxim's are not affected by the microcontroller's failure. But, in order to rely on them alone for safety, you would have to prove that the software is structured in such a way that every conceivable fault of the microcontroller as well as any software bug will prevent the watchdog from being toggled and, consequently, will lead to reset. This is next to impossible.

As you now understand, performance monitoring can add complexity to an otherwise simple design. Usually, designing a functional product represents no more than 30% of the engineering effort. Making sure it fails (it always fails) in a safe, predictable manner takes the rest of the effort. Ensuring that BIT covers all faults of complicated systems requires a testability analysis, which is outside the scope of this article. BIT coverage in devices such as this one can be analyzed as a part of the FMECA by careful review.

WHAT ABOUT SOFTWARE?

The circuit would have been easier to implement and with deeper test coverage by using two microcontrollers, each checking the other. The problem is software. Years ago, software was viewed as the proverbial pot of gold that would cut the cost of hardware to next to nothing. This expectation has not materialized, partly because of the lack of discipline and corner cutting prevalent among commercial software developers.

Recently, I watched some unfortunate person being psychoanalyzed on a TV show. The psychiatrist would say a word and the guy stretched on a couch replied the first thing that came to his mind. This made me realize that every time I hear "software," the word "paranoia" pops into my head. Today, developing software and certifying it for a safety-critical application is expensive. The current software standard DO-178B separates code development into five categories, A, B, C, D, and E, category A being the most demanding.

Systems in which software functions can be checked by hardware supervisors often can be certified to levels D, C, or B. Even sloppy, buggy software may satisfy safety requirements if monitored by hardware, albeit at a loss of versatility, which is the selling point for software usage (see Figure 3). Where there is a critical application performed and also monitored exclusively by software, level A is the only acceptable alternative.

To write, document, and certify to level A, the code for this hypothetical controller would require several thousand engineering hours. A simple, single line of code mod is not unusual to take several months to document and recertify. In addition, level A requires separation between design and test, that is, testing must not be performed by the people who designed the software. For more information, read "Joys of Writing Software" series (*Circuit Cellar* 120–123).

There are several alternatives when designing a 100% software-driven, redundant, safety-critical system. The simplest would be a like processor, like software design. Identical hardware channels running identical software are used, comparing each other. This is not a preferred method because you must show that no common mode failure is possible; there is no condition, be it wrong data, external interference, or fault, that can bring both channels down simultaneously. You would waste more time trying to prove this than if you pursued an alternative.

A more common method is a like processor, different software design. There are two similar hardware platforms, but the software for each is designed by a different engineer. Sometimes there are additional differences, such as the control channel performing calculations in 16 bits, and the monitor does it in 8 bits and uses the free time for communications. Often, to satisfy level A separation requirements, team A writes the controller and tests the monitor software and team B writes the monitor and tests the controller.

For the most critical applications where paranoia is the rule of the day, the different hardware, different soft-

ware approach is taken. It is assumed that a fault may exist in the microcode, and therefore, different processors are used. This may sound drastic, but when faced with a multimillion-dollar satellite's computer hanging up during the first orbit, going through the extra development effort is justified.

SV1	SV2	Pressure 1	Pressure 2	Pressure 3
off	off	1	0	0
on	off	1	1	0
on	on	1	1	1

Table 4—*The solenoid valve isn't considered part of your design responsibility. It is usually sufficiently reliable for shutting off the fuel supply. If you need to include it in the system, you may have to use two and perform diagnostics as shown in this table.*

For triple and more redundant systems, these approaches are equally applicable. The advantage of triple and higher redundancy is that devices can keep operating under failure conditions, as long as two out of three agree.

THE RESULTS

Now that you have modified the design after considering the reliability, FMECA, and FTA findings, let's look at the results. Let's discuss the functional block FMECA (see Figure 5). The first step is to look at the effect of the additional components on reliability prediction. Table 2 shows the updated design and includes improvements such as decrease of the junction temperature and application of duty cycle.

With the failure rate values calculated, you can proceed to perform FMECA (see Table 3).

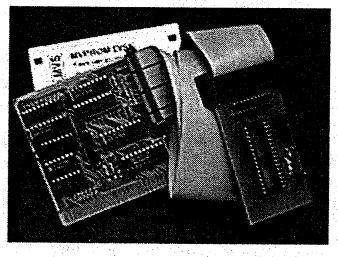
The important result is that all high criticality failures are monitored (see Figure 6). Again, the fault probability numbers for nodes are calculated by adding λ_p from the reliability prediction for every component within the functional block that could cause the given failure and multiplying it by 10^{-6} to obtain failure probability per 1 h. Where two failures are needed for the top event, the inputs are logicially ANDed (multiplied).

I should mention power supply failure mode A2, as well. For the output voltage to exceed 5 V and cause continuous heater operation, multiple faults would be required. Normally, FMECA and FTA are prepared on the basis of single faults. Logic AND gates exist for fault propagation, and the probability of multiple failures would be in the order of 10^{-12}. Because the power supply block can contain several dormant failures (i.e., the fuse and transzorb/crowbar circuit), you must treat the probabilities as logic OR. Fortunately, the monitor outside the power supply block will detect the excessive 5-V rail and switch off the SV via Q2.

A quick look at the FTA in Figure 6 shows that you exceeded the safety requirement by three orders of magnitude. But, there remains one other potential problem, the external valve. Its connection to the driver can short to the ground and cause continued energization of the valve. Or, the valve can be stuck in the open position.

$95 UNIVERSAL PROGRAMMER

Lowest cost, fastest, easiest to use product on the market! Does FLASH, EE, NVRAM, EPROM to 8m (27080). Adapters for micros & PLCC. Parallel port version for notebooks.



1meg $195 or 4meg $295
Battery backed memory
Hi-speed parallel interface
Windows/DOS software

NON-VOLATILE EPROM/FLASH EMULATOR

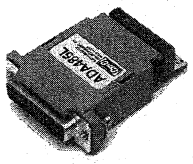
LOW COST LOGIC ANALYZER

Bus Tracer with IDE/ ISA/ LPT connectors for easy debugging or reverse engineering drivers. 64K 24bit $195, 512K 48bit $295

WWW.STAR.NET/PEOPLE/~MVS
MVS Box 850
Merr,NH 03054
(508) 792 9507
MVS
5yr limited warranty
FREE SHIPPING
Mon-Fri 10-6 EST

ASK ABOUT OUR LINE OF EMBEDDED CONTROLLERS AND SUPPORT PRODUCTS

RS232/RS422/RS485 Converters



CMC's low cost converters adapt any RS232 port for RS422 or RS485 operation. These converters provide your RS232 device with all the advantages of RS422 or RS485 including reliable high speed operation (up to 200 kbaud) and data transmission distances up to 5000 feet. Two AD422s can be used to extend any RS232 link up to 5000 feet. Completely transparent to the system; no software changes of any type are necessary.

RS232 TO RS422
- Converts bi-directionally between RS232 and RS422
- Use as a short haul modem
- Plug in and go. No software changes required

AD422 (Requires 9VDC) $79.00
AD422-1 for 110VAC 89.00
AD422L signal powered 84.00

RS232 TO RS485 2 wire
- Makes your RS232 port an RS485 port
- Supports up to 40 RS485 devices
- Automatically determines data direction.
- Signal powered version available

ADA485 (requires 9VDC) $79.00
ADA485-1 for 110VAC 89.00
ADA485L signal powered 84.00

RS232 TO RS485 4 wire
- Converts an RS232 port for use with RS422 or RS485 devices
- Supports up to 40 RS485 or RS422 multidrop devices
- Adds multidrop capability to RS232 devices
- Automatically determines data direction.

ADA425 (requires 9VDC) $89.00
ADA425-1 for 110VAC 99.00

Mention this ad when you order and deduct 5%
Use Visa, Mastercard or company purchase order code CC83

CMC Connecticut microComputer, Inc.
PO BOX 186, Brookfield,CT 06804 (203)740-9890

WWW.2CMC.COM Fax:(203)775-4595

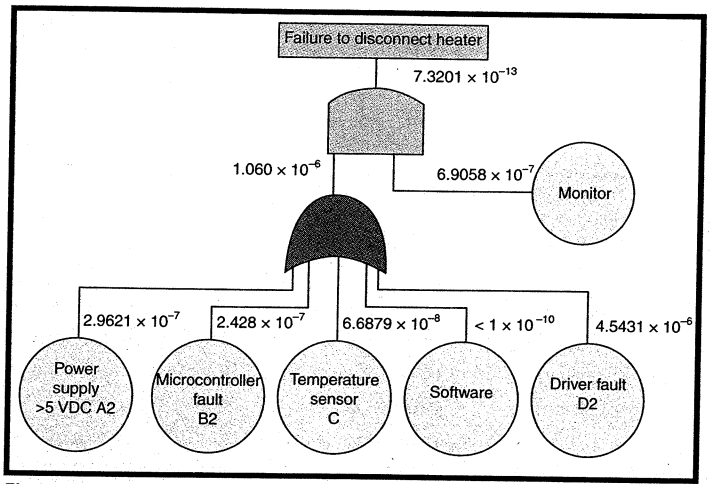


Figure 6—*The final fault tree analysis, which includes the monitoring circuit, proves that no single failure within the controller can cause a catastrophic event.*

The short to the ground problem can be addressed by careful wiring or, in a critical application, by using a high side driver or a dual high-low side interface. The mechanical failure of the solenoid valve is solved in many systems by using a high-quality valve with a filter on the input line to prevent dirt particles from entering. In critical applications, two valves are used. But this approach is expensive.

Both of the solenoid valves must have a totem pole driver. To monitor the valves' operation, you also need three pressure switches, one upstream of the valves (PS1), one downstream (PS3), and the third between them (PS2). The power-up BIT routine (P-BIT) energizes the valves as shown in the truth table (see Table 4) and reads the pressure to verify their operation. PS1 is there only to make sure the test routine is not performed without gas pressure, which would result in fault.

MAINTAINABILITY

The reliability prediction indicates that after you ship 10,000 units, you'll be ready to service at least two problems per day. You want to keep customers happy with a quick repair turn around time (TAT). You also want to keep the cost of service calls low.

Based on the complexity and cost of the controller, repair may be by replacement. The system is comprised of three subassemblies—the controller, temperature sensor probe, and solenoid valve. None of these is field repairable, so they are called line replaceable units (LRUs).

The maintainability analysis generates data showing the time needed to identify the faulty LRU, the time to replace it, and the time to retest the system and bring it up to speed again. This is called mean time to repair (MTTR), and identifies the tools, procedures, spare parts, and so forth needed for field repair. The analysis provides useful information for business planners and design engineers. For example, you may discover that a simple design change may eliminate uncommon tools otherwise necessary for the technician to carry. Or you may discover that the 5 min. required to replace the controller may have to be preceded by a 2-h system disassembly and followed by the same duration assembly.

Again, the most important aspect of the design is testability. Not only is it important in determining system safety, an effective BITE (built-in test equipment, the circuitry performing BIT) identifies the faulty LRU and displays it on the controller cabinet or transmits the data by a communications link. This reduces the MTTR. But, a 100% accurate BIT is nearly impossible to achieve. Usually 95% accuracy of fault isolation is acceptable; mean time between unscheduled removals (MTBUR) signifies the fault isolation accuracy. A controller with 10,000-h MTBF and 95% isolation accuracy will have 9,500-h MTBUR.

SUMMARY BENEFITS

In this two-part series, I approached a simple controller design from the perspective of reliability and safety. You learned how useful the reliability prediction, FMECA, and FTA become to an electronics designer. They help you create safe, robust designs, as well as provide insight into products' futures in terms of warranty, repairs, maintenance, and cost of ownership.

While covering this series, some questions were generated by introducing important subjects such as software development procedures and testability. These subjects need separate articles for a full discussion. For now, I want to reiterate that formal testability analysis is not only instrumental for BIT activity, but should be kept in mind while designing, even when there is no BITE present.

This applies equally to hardware and software. This requirement adds complexity to a simple design, but the alternative would be to prove the performance by analysis. Granted, there are functions that can't be tested, but the fewer the better. Proofs by analysis can be tedious, time-consuming, and quickly reach a dead end if conflicting engineering opinions come into play.

George Novacek has 30 years of experience in circuit design and embedded controllers. He currently is a general manager of Messier-Dowty Electronics, a division of Messier-Dowty International, the world's largest manufacturer of landing-gear systems. You may reach him at gnovacek @nexicom.net.

SOFTWARE

Reliability calculations are available on the *Circuit Cellar* web site.

REFERENCES

S.E.R. subcommittee, "Automotive Electronics Reliability Handbook," Society of Automotive Engineers, Warrendale, PA, February 1987.

P. Tobias and D. Trindale, *Applied Reliability*, Van Nostrand Reinhold, NY, NY, 1986. ISBN 0-442-28310-5.

U.S. Department of Defense, *Electronic Reliability Design Handbook*, MIL-HDBK-338, General Policy Series, no. 542.

U.S. Department of Defense, *Maintainability Program for Systems and Equipment*, MIL-HDBK-47OB, Washington D.C.: Government Printing Office, June 1995.

FEATURE ARTICLE

George Novacek

A Sure Thing

Guaranteeing 99.99999% Reliability

If you're a gambler, play the lottery, but if you want to take the gamble out of project design, then listen to what George has to say. Performance guarantees are an important factor in avoiding costly retrofits or redesigns after you've already built the prototype.

One of the challenges you face when designing an electric airplane or, for that matter, any other process control or robotic system, is the performance guarantee. This is something you must face early in the design process. Discovering that the system doesn't meet the performance guarantee after you built a prototype may not be too late to save the project, but will cost a lot of money in rework and late delivery. Investing a little time up front with paper and pencil will pay handsome dividends later. In this article, I will show you how to use reliability tools to your advantage during the concept stage of a new design.

RELIABILITY DATA

Reliability prediction, fault tree analysis (FTA), and failure modes and effects analysis (FMEA) are powerful design tools, but to use them effectively, you need solid data. Needless to say, your results will be only as good as your data. There are several excellent sources available. The best and most obvious source is your own data or the component manufacturer's records. Any QA (quality assurance) department worth its salt must have a database of product failures during manufacturing, testing, and in the field continuously updated. Often though, component manufacturers do not publish data for competitive reasons and your own records may be insufficient.

"Reliability Prediction of Electronic Equipment" (MIL-HDBK-217) is a military handbook that's a rich source of information. [1] You can download it free from www.dsp.dla.mil. The most recent revision is F, and you also should download Notices 1 and 2.

MIL-HDBK-217's attempts to mathematically model devices by their types. This is a mammoth task, given the variety of uses, environments, and manufacturing processes. It worked well during from the '60s to '80s, but with the explosion of microelectronics in the last decade and the unprecedented strides in their manufacturing process control, the MIL-HDBK-217 could not be updated fast enough. Nevertheless, when used judiciously, it remains an excellent tool.

Another useful and accessible tool is the Reliability Analysis Center (RAC) of the Department of Defense. The center has a web site that includes data books and other information. Unlike the MIL-HDBK-217, the information isn't based on mathematical modeling, but rather on field data obtained from manufacturers and users. You find the component you are interested in and receive a wealth of information not only about its failure rate, but also the types and distribution of failures, origin of the reports, and so on. This is the database your QA manager dreams of developing, if he only had access to all government suppliers' field data.

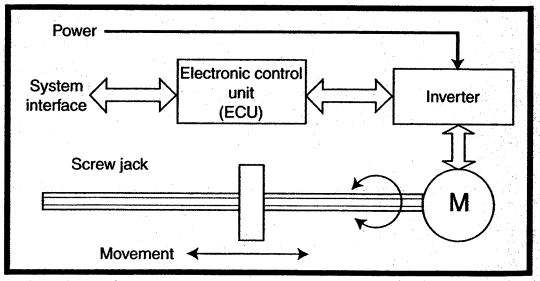


Figure 1—*A brushless motor drives a screw jack, which moves a mechanical arm.*

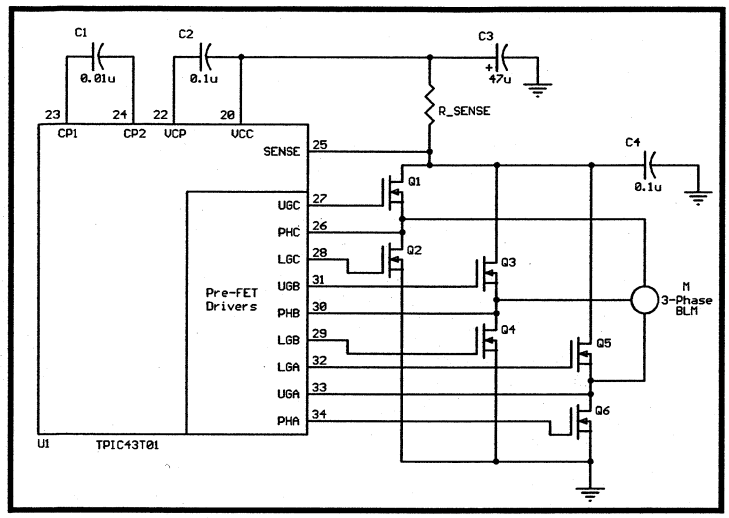


Figure 2—*A typical inverter can be built with power FETs and a control IC, such as this one from Texas Instruments. Many other ICs are available or you can create your own using an FPGA.*

Unfortunately, this tool is not free. It costs several hundred dollars, but is a bargain for the data it provides.

There is also commercial software available for people who cannot afford not to spend the high asking price for the tool of their trade. One of the better known, widely accepted tools is produced by Relex. You can obtain a database of electrical and mechanical components from the company's web site. And, the software will automatically generate the analyses for you and use different mathematical models, including MIL-HDBK-217.

99.99999% GUARANTEE

So, here's the problem: It makes no difference whether you are designing the electric airplane or a robotic system, your task is to design an electrically actuated motion system that moves some mechanical bits and pieces, be it control surfaces, brakes, or whatever. A failure of the system to move the parts won't be catastrophic, but will present enough problems for you to want to minimize the possibility of its occurrence. The customer has done the system hazard analysis and come up with the requirement that the probability of the failure must be less than 10^{-7}. In other words, the system availability must be better than $1 - 10^{-7}$, that's 99.99999%. Not a laughing matter!

This is where some analysis and simple calculations ahead of time can save you grief later. Figure 1 is a shows the system you are about to design. You will use a DC brushless motor because of its torque/speed characteristics, low maintenance requirements, and low EMI when compared with DC brush commutated motors.

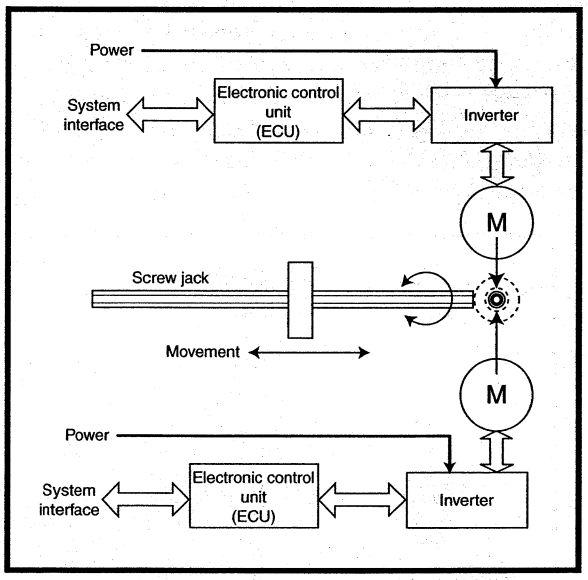


Figure 4—*Two motors provide a dual redundant drive by coupling through a planetary gear adder. The gear and screw jack remain single-point failures, so it is important that they have low failure rates.*

COMPONENT RELIABILITY

The first step will be to identify the individual system components and their reliability. The most important one is the motor, so let's start with that. Unlike most electronic components, as a result of wear, motor instantaneous failure rates are not constant but increase with time. Because the MIL-HDBK-217 failure rate model is based on a constant failure rate, you will develop an average failure rate for the motor operating over a time period known as its life cycle (LC). At the end of the life cycle, it is assumed that the motor will be replaced or overhauled. Thus, you can calculate the average failure rate:

$$\lambda_p = \left(\frac{\lambda_1}{A \times \alpha_B} + \frac{\lambda_2}{B \times \alpha_W}\right) \times 10^6 = \frac{\text{failures}}{10^6\ \text{h}} \quad [1]$$

where α_B is the Weibull characteristic life for the bearing and α_W is the Weibull characteristic life for the windings. These parameters depend on the operating temperature. Let's assume that the motor will operate in a room temperature environment from 25°C to 30°C. For this temperature, MIL-HDBK-217 states that α_B = 78,000 h and $\alpha_W = 8.9 \times 10^5$ h.

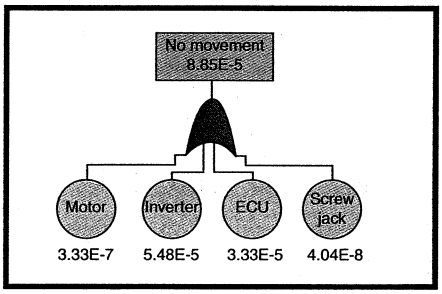


Figure 3—*The FTA shows you clearly that the system does not satisfy the specification requirement and helps you identify the cause. In this case, note that both the ECU and inverter's failure rates are higher than the required outcome.*

This mathematical model purposely does not take into account failure of commutators (brush or electronic). Brush commutators would have to be inspected and serviced regularly for this failure model to remain valid. As already stated, because this application requires a long life, maximum reliability, and minimum maintenance, you wouldn't consider using a brush commutated DC motor. But I hasten to add that the reliability of modern brush commutators is nothing to sneer at and you shouldn't dismiss this established technology.

For general application electric

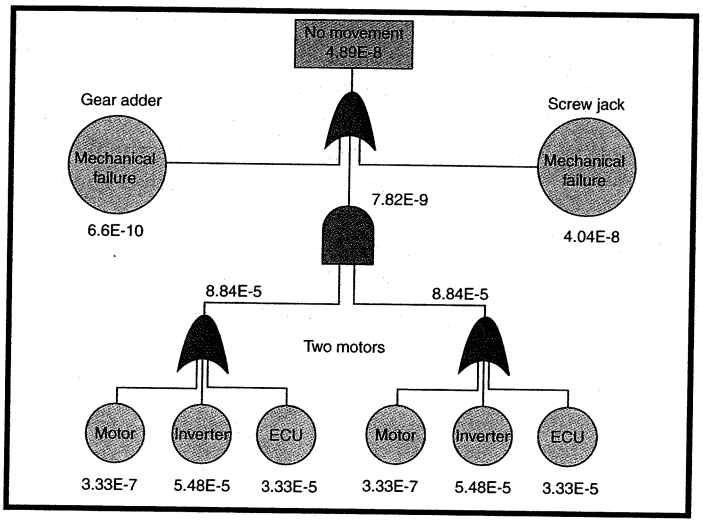


Figure 5—*Here's the FTA of the two-motor configuration shown in Figure 4. Notice the importance of the low failure rate of the gear and screw jack.*

motors, MIL-HDBK-217 shows constants A = 1.9 and B = 1.1. λ_1 and λ_2 are related to the life cycle (i.e., the expected operating life of the motor). The customer requires that the system last three years without the need for an overhaul. Although the entire system operates 8 h per day, your subsystem requiring 99.99999% availability will not be needed more than one third of this time. Therefore, you can calculate the LC to be 2,920 h, which results in $\lambda_1 = \lambda_2 = 0.13$. And then, plugging these values into Equation 1 results in:

$$\lambda_p = \left(\frac{0.13}{1.9 \times 78{,}000} + \frac{0.13}{1.1 \times 8.9 \times 10^5}\right) \times 10^6 \quad [2]$$
$$= \frac{\text{failures}}{10^6\,\text{h}} = 1.01 \times 10^{-6}$$

It is worth noting that the bearings have an order of magnitude greater effect on the motor failure rate than the windings, a fact I will revisit later. Because the motor will be required to operate no more than 0.33 of the system operating time, you can apply this duty cycle to its calculated failure rate and assume:

$$\lambda_{pr} = \frac{\lambda_p}{3} = \frac{1.01 \times 10^{-6}}{3} = 3.33 \times 10^{-7} \quad [3]$$

The other electrical components of the system comprise an inverter and an electronic control unit (ECU). The typical inverter is shown in Figure 2. It uses power FETs and a Texas Instruments' integrated circuit, TPIC43T01. Other power semiconductors, such as bipolar or IGBT transistors, can be used in place of the FETs. Similarly, there are numerous control ICs on the market. Or, you can design your own controller using a DSP or FPGA. Based on several different concepts with Hall effect diodes used for position sensing, component level calculation per MIL-HDBK-217 specification will yield an estimated failure rate of 2.01×10^{-4} for the inverter. After application of the 33% duty cycle, assuming that the power will be off when the function is not required, the final failure rate will be 5.48×10^{-5}.

The ECU will be a microprocessor-based embedded controller providing system interfaces, motion control, and most importantly, system diagnostics and failure detection. Similar systems I developed exhibit an MTBF better than 30,000 h in the harsh aerospace environment. For this article's calculation, you convert the MTBF into failure rate by calculating its reciprocal. The result equals 3.33×10^{-5}. The ECU can't take advantage of the duty cycle, because it will always be powered together with the rest of the system.

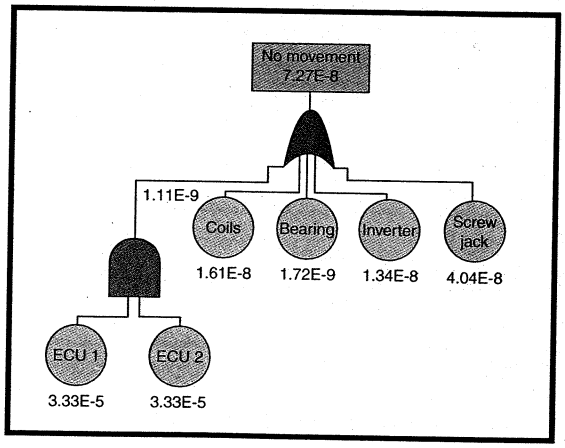


Figure 6—*The ECU is dual redundant, as is the inverter. As a result, the single motor system (brushless) satisfies the specification requirement.*

The motor will drive a screw jack as shown in Figure 1; if it fails, the whole function goes down. You do not supply this component. Make sure the customer understands this single-point failure and selects a component with failure rate roughly one order of magnitude better than the function needs. The screw jack selected has a failure rate of 1.22×10^{-7}. Fortunately, the duty cycle applicable here will bring it down to the acceptable 4.04×10^{-8}.

PUTTING IT TOGETHER

It's immediately obvious that the function cannot achieve the required 1×10^{-7} failure rate when the inverter alone is more than two orders of magnitude worse than the customer expects (see Figure 3). The system components, which include the motor, ECU, inverter, and mechanical linkage (screw jack), all feed into an OR gate, meaning that any one of these components failing will cause the function to fail. And the failures are additive, making the outcome almost three orders of magnitude worse than required.

What's the solution? The word for it is redundancy. By making the components redundant, both would have to fail for the function to fail. Their individual failures now feed into an AND gate. Mathematically this means that the failure rates multiply.

It is interesting to note that the three solutions proposed here provide similar failure rates. As a result, the best concept selection will not have to be based on the achievable reliability but on other design issues such as economics and practicality.

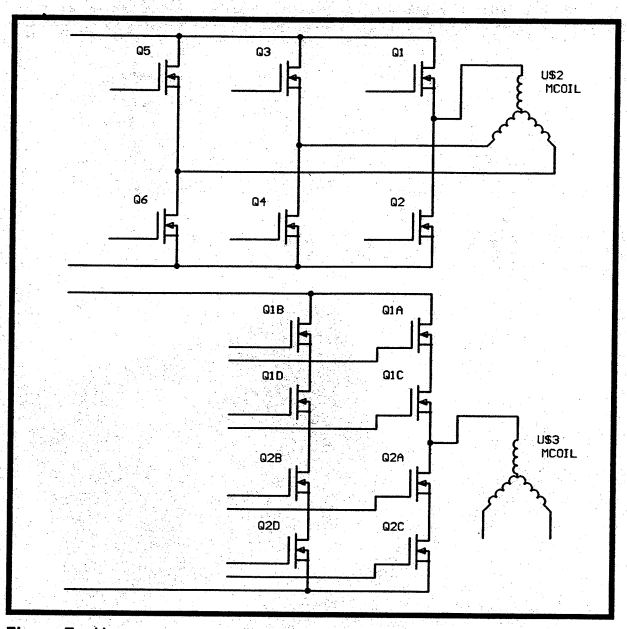


Figure 7—*Here you can see the business end of the inverter. Six power FETs originally needed to drive the three windings have grown to 24. Also, four independent driver ICs are needed.*

Figure 4 is the most obvious solution, frequently used in the past with brush commutated motors. The brush commutator represents a single-point, high-rate failure, which can't be easily fixed by redundancy. Therefore, two identical motors are coupled through a planetary gear assembly acting as an adder. This is analogous to a car differential drive with the motors attached instead of the wheels.

The FTA of this design is shown in Figure 5. The planetary gear coupler can be obtained with 2×10^{-9} failure rate, which is reduced to 6.6×10^{-10} by application of the duty cycle. Although simple, this configuration presents several, sometimes insurmountable, problems. First, it needs two motors. Their cost notwithstanding, the increase in size and weight may be prohibitive. The other problem is that the planetary gear is an adder. If one motor fails, the velocity of the screw jack will be cut in half, which may not be acceptable.

OTHER IDEAS?

The mathematical model for electric motors in MIL-HDBK-217 considers failure of the bearings and the windings. It doesn't take into account the different quality of bearings and windings you can achieve through process control nor does it fully account for different stress levels seen in brushless motors because the windings are stationary. A search through the RAC database reveals that the experienced failure rate of this kind of motor's bearings is 5.2×10^{-9} and the windings are 4.87×10^{-8}. With the application of the 33% duty cycle, these failure rates are reduced to 1.72×10^{-9}, and 1.61×10^{-8} respectively.

This means that the mechanical, failure-prone motor components, armature, and bearings exhibit failure rates much smaller than the permitted result. Therefore, they can be used in a single point of failure mode. It is the electronics in the ECU and inverter that are the problem and need to be redundant.

The FTA in Figure 6 shows the configuration that will do the job. Notice that two independent ECUs feed through an AND gate, thus achieving a 1.11×10^{-9} failure rate. This means that you must be able to determine which ECU is correct if there's a disagreement. This calls for a fail operative controller. The design of such a controller is outside the scope of this article, but I'll address it in the future. Also notice that the inverter's failure rate decreased dramatically, from 5.48×10^{-5} (using the 33% duty cycle) to 1.34×10^{-8}. How is it possible? Consider the simplified schematic diagram in Figure 7.

The failure distribution numbers in the RAC database state that the power FET failures are split roughly 50/50 between short and open circuit. This means that each power semiconductor device has to be replaced with four, such that no single failure can prevent the inverter from continuing to function.

So, while you can achieve the needed failure rate of 1.34×10^{-8}, the price you pay is the significantly higher component count and a more complex fault detection circuitry. Whether or not this is a practical approach is a matter of economics. For high-power, IGBT (insulated gate bipolar transistor) driven motors, which cost hundreds of dollars, it may be better to add a parallel set of windings to the stator (see Figure 8). The corresponding FTA in Figure 9 shows the result. The driver is now less complex and the winding dual redundancy helps lower the failure rate by about 30%.

THE NUMBERS GAME

You have seen how powerful and timesaving a simple reliability analysis can be when applied early. Used with common sense, and I must emphasize the common sense, it can save time, money, and frustration that always accompany rework and failures. Do not expect precision! Too many engineers make the mistake of confusing reliability prediction with accounting, not realizing that even accountants are creative.

The predicted failure rate is a number, usually reflecting the worst-case condition, originating from an imperfect mathematical model or statistical analysis that can rarely duplicate or account for all the working conditions your product will encounter. The sta-

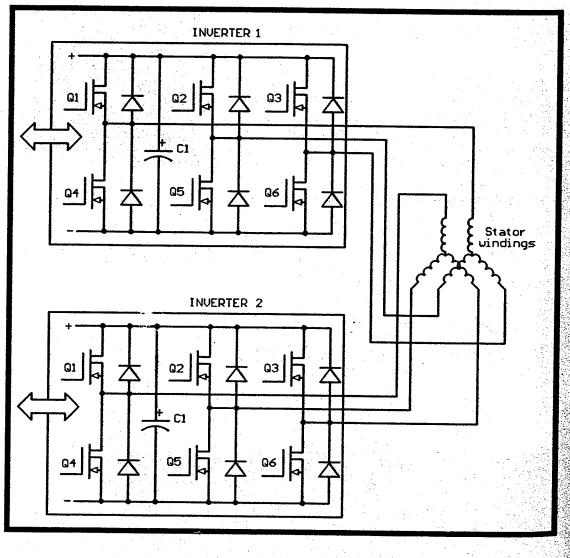


Figure 8—*This configuration saves 12 power drivers and requires a second set of stator windings.*

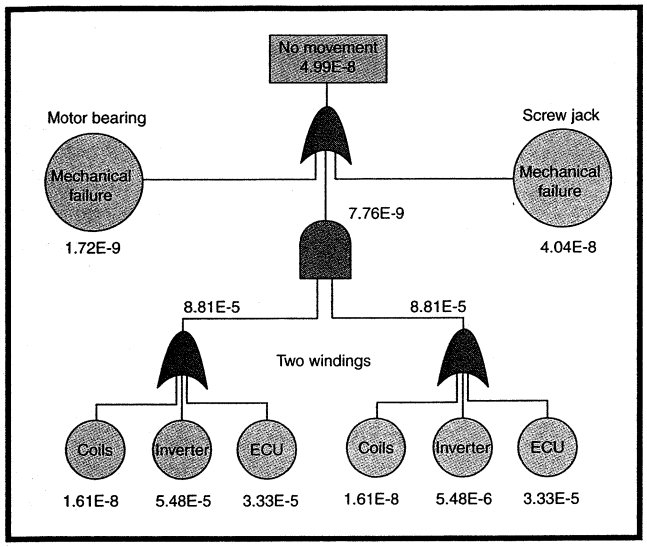


Figure 9—*The FTA shows the failure distribution of the two-windings configuration in Figure 8.*

tistical reliability prediction is an excellent tool for identifying potential problems and weaknesses early in the design process and for helping to model the system architecture to meet the intended specification. If I get within the same order of magnitude of the intended performance, I'm happy. I've seen too many (ignorant) customers excited about the analysis result being off by less than 1% and too many (equally ignorant) engineers wasting time by tweaking the numbers to achieve bureaucratic victory and "meeting the spec dead on."

It's a good idea to always keep the concept of the slide rule with its two decimal places of precision in mind. The imperfect world of engineering will rarely require more than that. Remember, the mere presence of 64 decimal places on your calculator display does not mean that the calculation based on your estimate will automatically acquire the same precision. So, make sure you don't lose your perspective by getting immersed in unimportant details.

WRAPPING IT UP

In the end, it is the performance that counts. No statistical analysis can change that. I have always seen the mature product reliability exceed the calculated value. The reason is not merely the conservative reliability model but the development process, as well.

Having identified weak parts, proper steps can be taken to avoid later problems. It is equally necessary to keep a record of all failures, analyze them, and take corrective action if necessary. In aerospace technology, this has an official name, Failure Reporting and Corrective Action System (FRACAS). Behind the long name is a common sense activity to close the loop between the user and designer.

With critical or large-volume products where the risk of field problems is not tolerable, accelerated testing is done as part of the reliability growth. The system is stressed until its weakest link fails. It is analyzed, corrected, and then stressed again. The purpose is to achieve not only the desired mature reliability quickly but also to have the reliability spread evenly across the product.

There is no point in having a sturdy, expensive design with one weak part causing failures. In fact, if such failures still meet the specification, it may be wise to degrade the rest of the components and reduce the cost.

The one thing I haven't talked about in this article is the power supply. Of course, if the power supply's reliability doesn't support the availability requirement of the function, there is nothing you can do about it. So, from the beginning, assume that the power will be available.

A rule of thumb is that, when it comes to DC motors, voltage gives you speed and current gives you torque. With the increasing power demands you put on DC motors, there is a practical limit for the current, beyond which it is advantageous to increase the voltage and obtain the torque by gearing down the motor's speed. Today, it is not unusual to see DC motors running at 300 VDC and spinning at over 20,000 rpm.

Although automotive systems are moving toward 42 VDC and avionic systems already use 28 VDC to reduce current, this is not enough for the high-power, 50-kW (unbelievably small) motors you encounter in modern servo systems. In a future article, I'll show how the power is generated and talk about some of the peripheral issues such as power quality.

George Novacek has 30 years of experience in circuit design and embedded controllers. He currently is the general manager of Messier-Dowty Electronics, a division of Messier-Dowty International, the world's largest manufacturer of landing-gear systems. You may reach him at gvovacek@nexicom.net.

REFERENCE

[1] U.S. Department of Defense, "Reliability Prediction of Electronic Equipment," MIL-HDBK-217F, Washington D.C.: Government Printing Office, 1995.

RESOURCES

G. Novacek, "Designing for Reliability, Maintainability, and Safety: Part 1—Getting Started," *Circuit Cellar* 125, December 2000.

G. Novacek, "Designing for Reliability, Maintainability, and Safety: Part 2—Digging Deeper," *Circuit Cellar* 126, January 2001.

SOURCES

Database of information
Reliability Analysis Center
Department of Defense
(888) 722-8737
(315) 337-0900
Fax: (315) 337-9932
rac.iitri.org

Software
Relex Software Corp.
(724) 836-8800
Fax: (724) 836-8844
www.relexsoftware.com

TPIC43T01
Texas Instruments, Inc.
(800) 336-5236
www.ti.com