FUNDAMENTALS OF BUILDING A TEST SYSTEM

System Maintenance

CONTENTS

Introduction

In a perfect world, systems would never fail. You would set them up, turn them on, let them run, and never think about them until you are ready to stop using them in 10, 15, or 20 or more years. Unfortunately this is not reality, at least not yet. Systems fail and sudden, unexpected failures can be costly. Although you cannot completely remove the risk of failure, even with the most well-thought-out plans, you can reduce it. Maintenance strategies can help you to manage this cost and reduce the risk of failure.

A maintenance program is critical to ensure the lowest total cost of ownership across the life of an automated test equipment (ATE) system. A system designed for maintainability coupled with a sound maintenance program helps:

- Maximize capital investment by maintaining functionality and extending the useful life
- Minimize downtime costs by managing logistics, scheduling, and sparing inventory

The goal of any maintenance program is to keep the system working correctly for as long as possible, and get it back to working quickly if it stops. And, by the way, do this as inexpensively as possible.

Concepts and Definitions

Maintenance is the activity of preforming service to keep a system functioning and repairing a system if it fails. Maintenance is divided into three areas: predictive maintenance, preventive maintenance, and corrective maintenance.

Maintainability is the ease in which maintenance can be conducted. Some industries refer to it as serviceability. The better the maintainability, the easier it is to control maintenance cost.

Predictive maintenance uses condition monitoring to detect a system failure before it occurs and is sometimes referred to in industry as condition-based maintenance. When a potential failure is predicted, maintenance activities are scheduled to service a system. These activities can extend the useful life and avoid unplanned downtime. Predictive maintenance activities are normally not scheduled until the need for maintenance is detected and result in planned downtime, which is typically much less costly than unplanned downtime. Planned downtime costs can be shared across many other systems receiving maintenance. The goal is to maximize the capital investment by using systems/components for as long as possible before a failure and minimize unplanned downtime costs. With the Internet of Things moving forward at an incredible pace, the concept of smarter machines that can monitor themselves and communicate to a network of other machines when they need maintenance is becoming the norm. Technology advances in sensors, embedded controllers, FPGAs, networks, and Big Analog Data™ analytics have made predictive maintenance easier and more cost-effective than ever. A measure of predictive maintenance is the downtime incurred; this time is referred to as the mean predictive maintenance time (MPdMT).

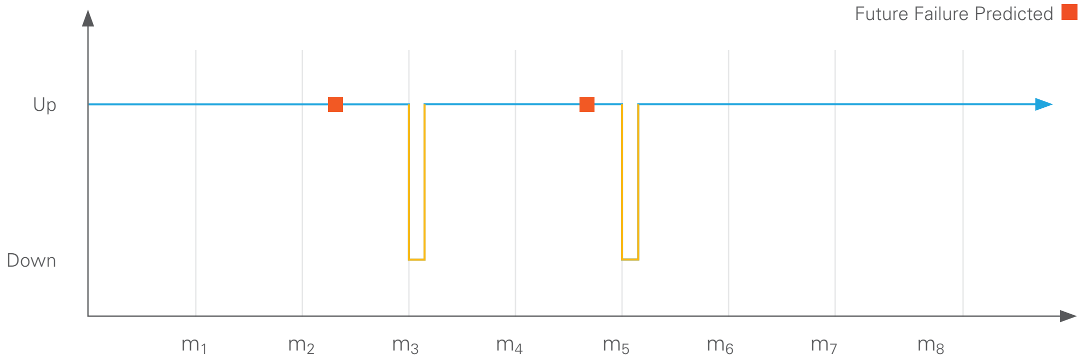

Figure 1. See predictive maintenance uptime and downtime over time. Predictive maintenance maximizes the use of your capital investment and minimizes downtime costs by lowering the frequency of downtime and converting expensive unplanned downtime to less expensive planned downtime, but requires failure monitoring equipment and prognostics software.

Predictive maintenance activities include:

- **Condition monitoring**—This ensures the system functions correctly, detects the onset of a failure, and identifies hidden failures in components or performance degredations that could lead to a system failure. With affordable embedded microprocessors and FPGA technology, built-in self-tests and conditioning monitoring techniques are commonly used. This is sometimes referred to as prognostics and health management (PHM) or system health monitoring. The concept is to detect performance changes and hidden failures in the system before they cause a much more serious system failure.

  Today, most cars have automated engine health monitoring that detects issues and flashes the check engine light, hopefully, in time to have the engine serviced before it is permanently damaged. A test system may monitor temperature, fan speed, memory usage, test times, measurement accuracy, count relay operations, and so on.

- **Servicing system components**—This helps to slow down wear and increase the useful life of the system.

  Some car tires have sensors that check the air pressure. Improper air pressure can shorten the life of tires and affect gas mileage performance. If a test system is used in a dusty environment, it may need to clean the dust from the air filters and the inside of the enclosure so it will not overheat and shorten the useful life of the electronics. Monitoring the internal temperature or airflow of the system can inform you to when you may need to clean dust filters.

- **Replacing system components**—Components are replaced before they fail to avoid unplanned downtime.

  A test system may use relays to switch signals for testing the device under test. Depending on the electrical load switched, a relay lasts for only an estimated number of operations. Therefore, monitoring the number of operations and replacing the relay modules before they fail is usually more cost-effective than waiting until a failure happens and experiencing an unplanned outage.

- Calibrating to compensate for drift—The purpose of a measurement system is to provide trusted measurements. If measurements are untrustworthy, then the system is functioning incorrectly.

  Most test systems contain electronics that need calibrating at some interval. If cutting-edge technology is used, however, the calibration interval may not be well understood, yet. Therefore, monitoring the measurement drift may be advised to understand when to properly schedule calibration maintenance.

- Verifying—This ensures the system functions correctly before bringing it back online. If it were brought back online only to malfunction, downtime would increase.

- Bringing the system back online—This must always be considered because, for some applications, it is not a trivial task.

  For example, if the test is part a manufacturing process, to bring this system back online may require stopping the line and resynchronizing the tester with the production flow.

Preventive maintenance is the activity of servicing a system to prevent a system failure and extend useful life. Preventive maintenance activities are normally scheduled and result in planned downtime. Planned downtime costs can be shared across many other systems receiving preventive maintenance. The goal is to minimize unplanned downtime costs. A measure of preventive maintenance is the downtime incurred; this time is referred to as the mean preventive maintenance time (MPMT).

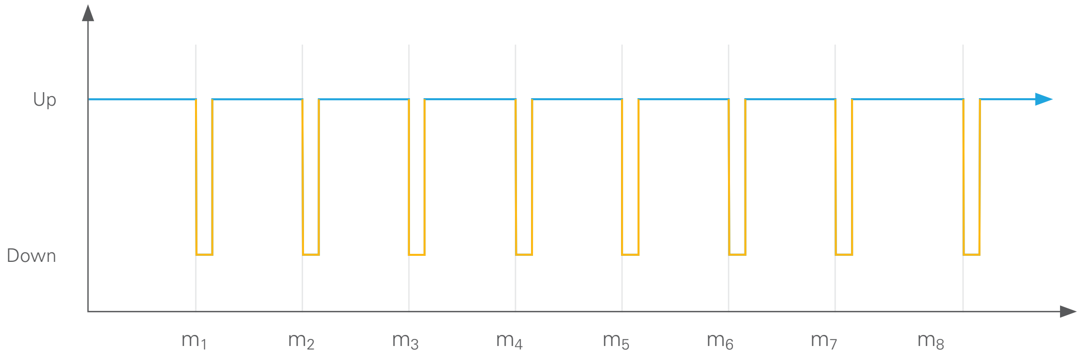

Figure 2. Preventive maintenance does not always maximize the use of your capital investment, but it helps to minimize downtime cost by avoiding expensive unplanned downtime.

Preventive maintenance activities include:

- Servicing system components—This helps to slow down wear and increase the useful life of the system.

  This is why a car's engine oil needs to be changed regularly. Test systems usually have complex software programs running in them that can have hidden resource leaks and or faults that eventually cause a system failure. A simple system reboot can refresh the software to a good-as-new state. If a test system is used in a dusty environment, it may need to clean the dust from the air filters and the inside of the enclosure so it will not overheat and shorten the useful life of the electronics. If the temperature and/or airflow cannot be monitored, then scheduling regular maintenance may be required.

- Replacing system components—Components are replaced before they fail to avoid unplanned downtime.

  The tires or break pads on a car are replaced at a certain mileage to avoid a failure that may cause an accident or strand the driver. A test system may have connector pins to test the device and they tend to wear out after 100,000 connections. If 50 devices are tested per hour, then the connector should last about 2,000 hours or 83 days before it wears out and fails. Preventive maintenance should be scheduled about every 80 days to replace the connectors. Replacing before a failure is usually more cost-effective than waiting until a failure happens and experiencing an unplanned outage.

- Calibrating to compensate for drift—The purpose of a measurement system is to provide trusted measurements. If measurements are untrustworthy, then the system is functioning incorrectly.

  Most test systems contain electronics that need calibrating at some interval.

- Verifying—This ensures the system functions correctly before bringing it back online. If it were brought back online only to malfunction, downtime would increase.

- Bringing the system back online—This must always be considered because, for some applications, it is not a trivial task.

  For example, if the test is part a manufacturing process, to bring this system back online may require stopping the line and resynchronizing the tester with the production flow.

Corrective maintenance is the activity of repairing a failed system to restore it to a functioning state. Corrective maintenance activities are usually not scheduled and result in unplanned downtime. The goal is to maximize the capital investment by using systems/components for as long as possible before a failure and after a failure to minimize unplanned downtime costs. A measure of corrective maintenance is the downtime incurred by a failure; this time is referred to as the mean time to repair (MTTR).

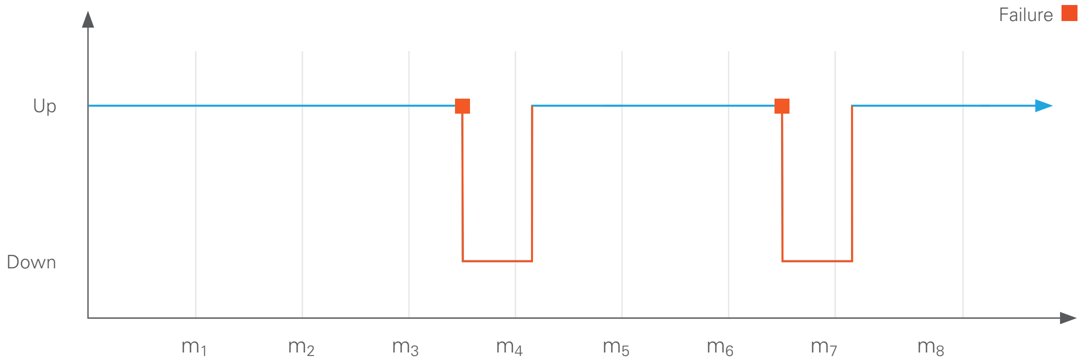

Figure 3. See corrective maintenance uptime and downtime over time. Corrective maintenance maximizes the use of your capital investment, but does not minimize the cost of downtime because it is unplanned. You can take steps to minimize the duration of the unplanned downtime or MTTR.

Corrective maintenance activities include:

- **Detecting**—Detecting a system failure as soon as possible minimizes costly unplanned downtime and possibly prevents damage to other components of the system and/or other systems that are used in the same process.

  Pressure sensors in a car can detect a drop in oil pressure as soon as possible to alert the driver and prevent permanent damage to the engine. Maybe the oil pump failed or the oil level is low because of a leak. It is much less expensive to repair an oil pump or seal a leak and add more oil than buy a new engine. For ATE systems, electronics may fail that can affect critical measurements and cause incorrect test results. If the failure took time to detect, a company could unknowingly ship bad products to customers, or a cooling fan could fail and chassis temperature may rise to a level that damages some of the electronics.

- **Diagnosing and isolating**—Diagnosing and isolating a failure correctly after it is detected can minimize unplanned downtime and save money by helping operators and maintenance personnel find and repair the correct component quickly.

  Automotive mechanics have sophisticated diagnostic equipment to help them diagnose problems efficiently and effectively. This saves time and money by lowering the risk of repairing or replacing the wrong component. The same can be said for complex ATE systems—hours and even days can be spent diagnosing a problem without proper diagnostic tools.

- **Repairing**—The system is repaired by repairing or replacing a failed component. The unplanned downtime is greatly impacted by having spares available. Depending on the application, environment, and skill level of personnel, having a spare system or spare components located nearby may or may not be cost-effective or practical.

  Most would not drive across the country without a spare tire in the car, but might if they need to drive only a few city blocks.

  A sparing strategy is essential to control costs. It is important to consider questions like, will the spares be kept on-site or in a nearby service center? Will you pay for the supplier to send an advanced replacement unit from the factory, or just wait until the failed unit is repaired and returned? The cost of unplanned downtime drives the answers. The number of units, the unit's mean time between failure (MTBF), and the time it takes to replenish the pool of spares determines the number of spares needed. Some companies provide levels of sparing services to assist with estimating the number of spares needed, helping with logistics, and managing sparing costs.

- **Verifying**—This ensures the system functions correctly before bringing it back online. Without this step, the system may still be functioning incorrectly and just cause more unplanned downtime.

  Imagine having the breaks on a car repaired, and then driving the car at high speeds on a freeway without first testing the breaks to verify they actually work.

- **Bringing the system back online**—This must always be considered because, for some applications, it is not a trivial task.

  For example, if the test is part a manufacturing process, to bring this system back online may require stopping the line and resynchronizing the tester with the production flow.

Figure 4. The cost of unplanned downtime is typically much more expensive than planned downtime as shown by the comparison of uptime and downtime.

Design for Maintainability

A system's design greatly affects the ability to achieve an effective, high-quality, and controllable maintenance program. When designing an automated test system for higher maintainability, consider the following best practices and guidelines.

Self-Test and Monitoring

Self-test and monitoring are essential for reducing downtime, planned and unplanned. Designing self-test and monitoring capabilities into the system from the ground up is key to having efficient and effective health monitoring, failure detection, failure diagnostics and isolation, and system verification.

Modular Designs

Modular designs simplify activities and reduce time associated with servicing, replacing, repairing, and calibrating system components. They also improve system diagnostics and failure isolation, saving valuable time during an unplanned outage. In addition, they reduce costs associated with spares. Instead of keeping several complete systems in the sparing pool, you can keep components, subsystems, or modules. Components usually have different failure rates—the components with lower failure rates need fewer spares, whereas those with higher rates need more.

A powerful strategic benefit of a modular design is that you can easily address new test requirements because of the upgradeability of the system. New technology and the requirements to test it are not just continuously changing, but the continuous change is accelerating. You would probably need to retire or scrap a monolithic nonmodular-designed test system and build a completely new test system every few years, which can become expensive. A modular-designed test system, however, can be much more flexible and upgradeable and cheaper in the long term. Its ability to easily accommodate new requirements is powerful. Instead of building a whole new test system, you may need to change only a few subcomponents. Meeting the new test requirements can be as simple as adding or exchanging a couple of modules, upgrading the controller for powerful processing capabilities, and/or modifying the software.

Standardization

Standardization can greatly reduce costs because it simplifies logistics and reduces the number of spares, amount of maintenance tools and equipment needed, and training costs.

For example, some airlines employ 10 or more types of aircrafts in their fleet. Southwest Airlines, however, uses just one—the Boeing 737. This results in cost-savings. Mechanics need to be trained on and need spare parts inventory for only one type of airplane. They can swap out a plane at the last minute for maintenance. The fleet is totally interchangeable. All onboard crews and ground crews are already familiar with it. And, there are no challenges in how and where the planes can be stored, because they're all the same shape and size.

Standardization greatly helps to control the maintenance process. A well-controlled process is repeatable and predictable, thus designing a system with a maintenance process that can have only one way to complete the task is essential. If the Southwest Airlines maintenance crews all used different tools and conducted maintenance tasks differently, then each crew would produce different levels of quality and take different amounts of time to do the work, which makes it difficult to control and manage maintenance costs.

Simplicity

Keep it as simple as possible to operate and maintain. In other words, make it easy to do the right things. This reduces the amount of documentation and training required, improves the consistency of the work, and decreases the time needed to conduct maintenance.

Environment and Human Factors

Always consider the environment and human factors. For example, if the system is used in a dusty environment, it may need dust filters on the air vents. How easy will it be to service the filters? Does the system need castors so you can move it around for maintenance? If so, make sure they are appropriate for the weight and terrain. What is the skill level of the operators and maintenance personnel and how much training do they need? Can you design hardware and software interfaces in a user-friendly way?

DESIGN GUIDELINES	PREDICTIVE	PREVENTIVE	CORRECTIVE
Self-Test and Monitoring	▪ Condition monitoring ▪ Verifying functionality	▪ Verifying functionality	▪ Detecting failures ▪ Diagnosing and localizing failures ▪ Verifying functionality
Modular Design	▪ Condition monitoring ▪ Servicing ▪ Replacing ▪ Calibrating ▪ Verifying functionality	▪ Servicing ▪ Replacing ▪ Calibrating ▪ Verifying functionality	▪ Detecting failures ▪ Diagnosing and localizing failures ▪ Repairing ▪ Verifying functionality
Standardization	▪ Condition monitoring ▪ Servicing ▪ Replacing ▪ Calibrating ▪ Verifying functionality ▪ Improving consistency of work	▪ Servicing ▪ Replacing ▪ Calibrating ▪ Verifying functionality ▪ Improving consistency of work	▪ Detecting failures ▪ Diagnosing and localizing failures ▪ Repairing ▪ Verifying functionality ▪ Improving consistency of work
Simplicity	▪ Lowering documentation and training costs ▪ Improving consistency of work	▪ Lowering documentation and training costs ▪ Improving consistency of work	▪ Lowering documentation and training costs ▪ Improving consistency of work
Environment and Human Factors	▪ Lowering frequency of predictive maintenance events and/or the MPdMT ▪ Reducing human errors ▪ Improving safety	▪ Lowering frequency of preventive maintenance events and/or the MPMT ▪ Reducing human errors ▪ Improving safety	▪ Lowering failure rates and/or the MTTR ▪ Reducing human errors ▪ Improving safety

Table 1. This high-level summary shows how each design guideline benefits each maintenance approach.

Maintenance Strategies

Which approach should you use? Predictive strategies wait until a potential future failure is detected and then schedule service or replacement at a convenient time. Preventive strategies proactively service, replace, and/or calibrate system components on a regular scheduled interval to minimize the risk of failure and the cost of unplanned downtime. Corrective strategies wait until a component fails to maximize the usage of the capital investment and repair it as quickly as possible to minimize the cost of unplanned downtime, or minimize the MTTR. For each strategy, you can do it yourself, work out a service agreement with the suppliers, or do nothing and hope for the best when a failure happens, which is not recommended.

Here, see a combination of techniques that help explain which maintenance strategy is best to use for different subsystems or components. The approaches discussed are condition monitoring feasibility, reliability centered maintenance (RCM), and cost of failure analysis. RCM is based on having an understanding of the affect of runtime on the failure rate of system components and the cost of component failures. The failure rate as a function of runtime is shown in the three graphs below. Each graph depicts characteristics for different types of components. There are more scenarios than these three, but these are common ones that help demonstrate how RCM works.

Figure 5 shows the failure rate increasing overtime. In this situation, the component's failure rate may appear constant at first but starts to enter wear out well before the intended service life of the system. In other words, the useful life of the component is significantly shorter than the length of time the system will be in service. This is probably the most intuitive scenario because mechanical components like fans, connectors, electromechanical relays, solid-state hard drives, batteries, the calibration of electronics, and so on have this trend. After each preventive maintenance event, the failure rate is lowered back to a "good-as-new" level, thus restoring the reliability of the system.

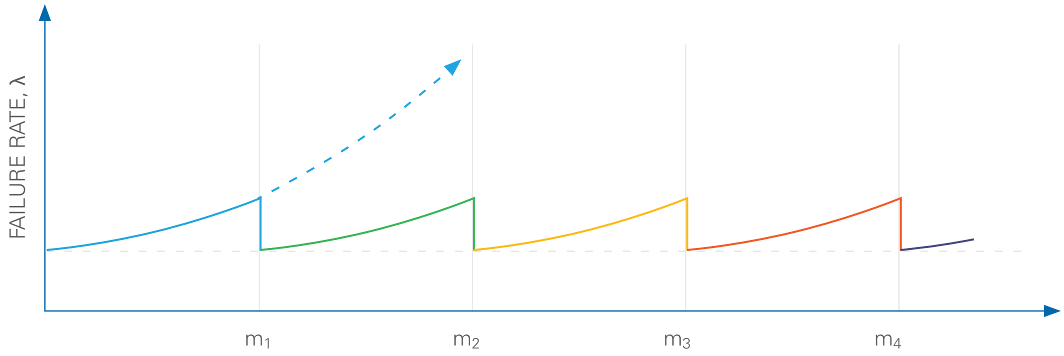

Figure 5. Preventive maintenance lowers the failure rate back to a "good-as-new" level at each maintenance event when the failure rate is increasing.

Figure 6 shows the failure rate remaining constant over time, sometimes called the steady-state failure rate. In this situation the component should not start to wear until well past the intended service life of the system (this does not include calibration). In other words, the useful life of the component extends well beyond the length of time the system will be in service. This is a typical scenario for electronic components such as ICs, resistors, ceramic capacitors, diodes, inductors, and so on that are in useful life. Modern electronics typically have a useful life well beyond 10 to 15 years. For all practical purposes, they do not wear out before the test system is obsoleted.

After each preventive maintenance event, the failure rate is not changed, so there is no benefit to replacing a component before it fails. Mathematically, this failure rate is treated as a "random chance". Therefore, replacing an older functioning component with a new component does not improve the system reliability.

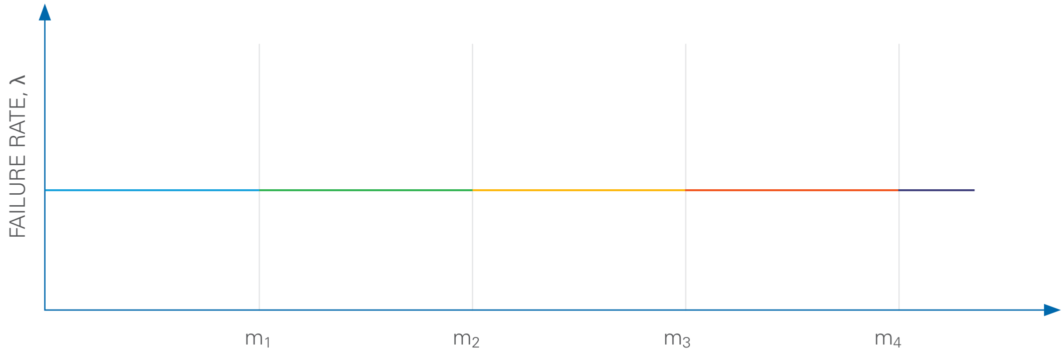

Figure 6. Preventive maintenance has little to no effect on the failure rate at each maintenance event when the failure rate remains constant over time.

Figure 7 shows the failure rate decreasing over time. This is probably the least intuitive scenario, but software and complex computer systems can exhibit this characteristic. Performing major upgrades to software and firmware or adding new features, new technology, and so on can introduce defects (bugs) that increase the probability of a system failure. After each preventive maintenance event, the failure rate is raised to a higher level, thus decreasing the system reliability. However, situations arise where you must upgrade software, such as OS updates or hardware obsolescence.

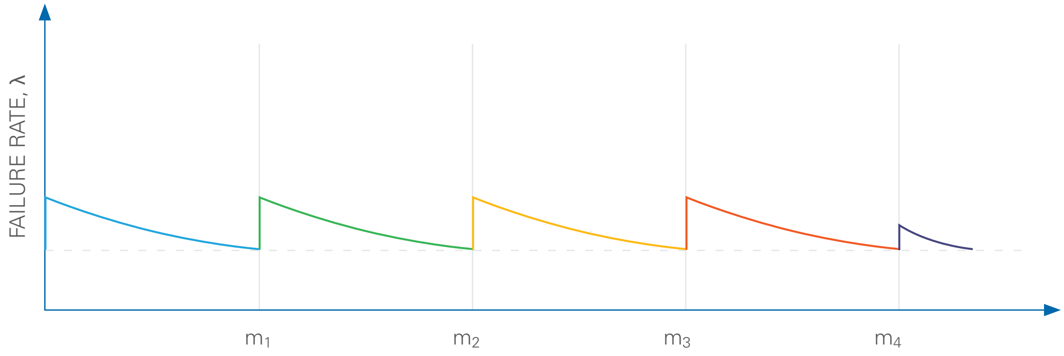

Figure 7. The failure rate decreases over time and the preventive maintenance actually raises the failure rate at each maintenance event.

In addition, there are situations when there is insufficient data to know whether the failure rate is increasing, constant, or decreasing over time. This may be the situation for new products, technologies, or designs. Using a predictive maintenance strategy of monitoring for failures over time can provide good insight into what the situation for a component might be if the cost of monitoring is effective compared to the cost of a failure. Even if a trend is not established, a predictive strategy usually maximizes your capital investment and minimizes downtime costs.

When using this approach to develop a maintenance strategy for a complete system, you can break down the system into subsystems and/or components, and then evaluate each component to see which maintenance strategy is best. The following are some good guidelines to work with:

- Can the onset of a component failure be detected before it causes a system failure?
  - Is it cost-effective to implement condition monitoring for this component failure, considering the cost of failure of a corrective maintenance event and the extra planned downtime from a predictive maintenance event?
- Is the failure rate of this component increasing, continuous, or decreasing over time or do you know?
  - Is the failure critical and the cost of a failure high?

The diagram below shows a decision flow chart to help you choose the best strategy for each component and failure mode of the system. The flow chart, however, should not override good human judgment.

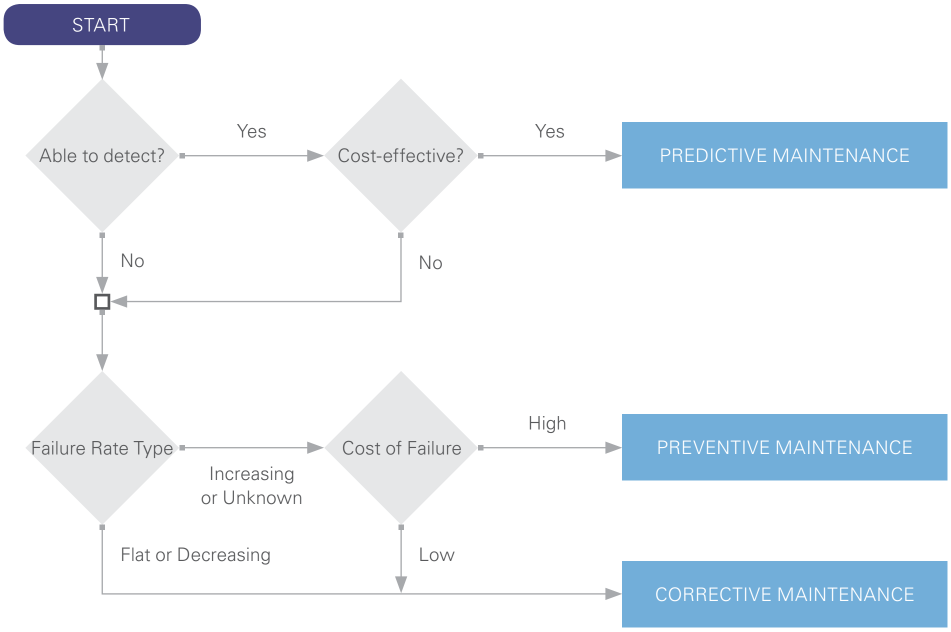

Figure 8. Maintenance Strategy Decision Flow Chart

Example ATE System

An ATE system based on PXI Express is made up of basic components or subsystems that can each be broken down into smaller components for a maintenance strategy of their own.

Chassis

The chassis backplane can be challenging to monitor for potential failures. It has a constant failure rate and should have a service life of 10 to15 or more years. The electronics are basically all digital and do not require calibration. A corrective maintenance strategy or a "run to failure" is the best approach.

The chassis power supply in this example does not provide monitoring capabilities. Power supplies typically use larger liquid filled capacitors and some may have their own cooling fans. These components have a typical service life of around seven to 10 years, depending on load and environmental conditions. A predictive or preventive maintenance strategy is a good approach.

The chassis fan speed and the chassis temperature can be monitored. If the speed starts to slow down or, if the chassis temp starts to increase, a warning can be sent and maintenance can be scheduled at a convenient time in the near future. A predictive maintenance strategy works well.

Controller

The controller's integrated circuits and electronic components (excluding hard disk and RAM) may provide tools to monitor for potential failures. For this example, it would require a lot of development time to implement these features and not be cost-effective. The average service life of a CPU is five to 10 years; thus, depending on how long you need the test system to be in service, a corrective maintenance "run-to-failure" approach or preventive maintenance approach may be the best.

The controller's RAM has an error correction code (ECC) that automatically runs and the amount of errors that are found and corrected can be monitored. If the frequency of these errors continue to increase, time to replace the RAM may need to be scheduled. RAM does not require calibration. A predictive maintenance strategy is the best approach.

The controller's hard drive in this example is a solid-state hard drive (SSD) that monitors the number of reads and writes. SSDs have only a certain number of reads and writes before they wear out. Thus, as the number of reads and writes approach the wear out numbers, the SSD should be scheduled for replacement. SSDs do not require calibration. A predictive maintenance strategy is the best approach.

Software has some unique characteristics—it does not wear out and is unaffected by the environment. It fails only because of design defects or bugs. Resource leaks, like memory usage and fragmentation, can be monitored; however, many faults cannot be seen before the crash. The wonderful aspect about software is that because it does not wear out, a simple reboot of the system is like starting fresh and good as new. This fixes everything until the bug raises its ugly head again. Therefore, a preventive maintenance strategy of rebooting the software once a week or once a month may solve many problems. A more challenging aspect to software reliability is upgrading. Software requires upgrades occasionally as new features are required, compatibility with other software packages is required, or perhaps a patch to fix a bug is needed. The challenge is that every time you introduce new software, it changes the ecosystem and may introduce more bugs. You don't know until you try. This dynamic makes the risk of failure for software go up immediately after a software upgrade, and then settle down after some runtime. The upgrade maintenance approach most commonly used for software is to delay an upgrade until it is necessary.

Instrument Modules

The integrated circuits on instrumentation modules can be challenging to monitor for potential failures. They have a constant failure rate and should have a service life of 20 or more years. The analog electronics can drift over time, thus they require calibration. A preventive maintenance strategy to address calibration is required to address drift. Many calibration labs can run a final verification test on the module after calibration to prove all is well. This test does a good job catching other electronic components that have failed or are failing. But no test is perfect and a corrective maintenance strategy or a run-to-failure approach may be best for some of the other failure modes of the electronics. There, a combination strategy is the best approach.

Switch Module

The switch module's base board is primarily made up of integrated circuits that usually do not have tools to monitor the health of the electronics. They have a constant failure rate and should have a typical service life of 10 to 15 or more years. The electronics are basically all digital and do not require calibration. A corrective maintenance strategy or a run-to-failure approach is best.

The switches' electromechanical relays have tools that monitor the number of operations. Relays have only a certain number of operations before they wear out, depending on the electrical load that is switched. You can estimate the number of switches by using data and formulas that the manufacturer provides. Thus, as the number of operations approaches the wear out numbers, the switch module should be scheduled for replacement. Switch modules do not require calibration. A predictive maintenance strategy is the best approach.

Cables

Fixed cables are basically connected and never disconnected, or reconnected so infrequently that it does not matter. Fixed cables hardly ever fail except form vibration or human abuse. The failure rate is constant and very low. A corrective maintenance strategy is the best approach.

Dynamic cables are connected and disconnected frequently and wear out after a certain number of reconnects. The failure rate is increasing over time and detecting a potential failure may not be easy, but it may be estimated. The number of reconnects may be understood by the manufacturer and is worth asking about. If the average number of reconnects is known and you know how many reconnects there will be per hour, per day, per unit, and so on, then you can schedule preventive maintenance. A preventive maintenance strategy is the best approach.

SUBCOMPONENT	PREDICTIVE	PREVENTIVE	CORRECTIVE
Chassis Backplane	—	—	√
Chassis Power Supply	—	√	—
Chassis Fans	√	—	—
Controller Mother Board	—	√	—
Controller RAM	√	—	—
Controller Solid-State Drive	√	—	—
Controller Software	—	√	—
Instrument Module	—	Calibrate	√
Switch Module Base Board	—	—	√
Switch Module Relays	√	—	—
Fixed Cables	—	—	√
Dynamic Cables	—	√	—

Table 2. You could use this maintenance strategy for each major component of an ATE example based on PXI Express. Note that the best strategy for each component is dependent on the unique situation for your application.

Conclusion

Each predictive, preventive, and corrective approach has its benefits, challenges, and appropriate situation. In most situations, the greatest expense associated with maintenance is the cost of unplanned downtime (the cost of a failure). Converting unplanned downtime to planned downtime through the use of condition monitoring and prognostics is usually advantageous.

Every year, condition monitoring equipment, networks, servers, and Big Analog Data™ analytics continue to decrease in cost and increase in performance, thus industry is trending toward smarter equipment and more predictive maintenance. For the situations when unplanned downtime is unavoidable, a good sparing and repair strategy is key to minimizing and managing maintenance cost.

A system designed for maintainability from the ground up coupled with good maintenance strategies will help you manage costs and reduce the risk of failures that lead to expensive unplanned downtime. This lowers the cost of maintenance, which lowers the total cost of ownership. Self-tests, modular designs, standardization, simplicity, and environmental/human factors are fundamental building blocks when designing for maintainability.

Appendix: Cost of Maintenance

Many companies base purchasing decisions for test equipment primarily on the price and do not consider the cost of deploying, operating, and maintaining the equipment. And they even less frequently consider the cost of equipment downtime. The cost of downtime (or failures) and maintenance over the service life of a test system can be much greater than the purchase price, frequently reaching two to three times more. The largest culprit is the cost of downtime or a failure. This is why maintenance programs exist and the maintainability of a system is becoming more important every day.

This appendix provides a straightforward total cost of maintenance (TCM) model that can be used to estimate the potential downtime and maintenance costs of a system over its service life. Calculating the TCM of a test system can become very tedious and complex quickly. This model provides a sufficient estimate at a level of complexity and detail that is adequate and manageable for most applications.

Total Cost of Maintenance (TCM)

$$TCM = CD + M$$

CD = *Cost of Unplanned and Planned Downtime*

M = *Cost of Maintenance*

You can measure a maintenance program's return on investment (ROI) by comparing maintenance dollars (M) invested to the reduction in downtime dollars (CD) spent over the service life of the system or to the overall reduction of TCM dollars over the service life of the system. Some companies combine the cost of planned downtime with the cost of maintenance and compare this only to the cost of unplanned downtime, because their main focus is to avoid unplanned downtime and failures. Each company may have its own way to estimate TCM and the ROI of maintenance, depending the metrics a company would like to track.

Cost of Downtime (CD)

Downtime costs can sometimes seem like “funny” money because some companies find them difficult to estimate. But the cost of downtime is very real. There are two types of downtime: planned (scheduled) and unplanned (unscheduled). The goal of a maintenance program is to minimize all downtime and convert as much unplanned downtime to planned downtime as financially feasible.

Unplanned downtime is always the most expensive, because it takes place when you need the equipment. It is never at a good time and can result in lost revenue from loss of production, product loss, collateral damage to other equipment, labor loss (the labor force may have to “‘sit’ around and” wait for the system to be repaired), and other logistical costs that are situation dependent. Some manufacturing companies estimate their cost of unplanned downtime to be around $8,000 per hour. Petrochemical, power, and transportation companies often estimate much higher cost per hour. This cost is different for various products, situations, companies, and industries. Time is money; this is why corrective maintenance plans with sparing strategies are put in place to minimize the mean time to repair (MTTR) of a failed system.

Planned downtime is costly as well, but less expensive than unplanned downtime because it is scheduled for times that will have the least impact on production, minimizes product loss, minimizes the risk of collateral damage to other equipment, results in no labor loss, and minimizes the cost of logistics (trained people, tools, and parts are on-site and ready to perform the maintenance). Planned downtime can be shorter than an unplanned outage and is shared across many other systems that need maintenance. Because unplanned downtime is usually much more expensive than planned, many companies put into place predictive and preventive maintenance plans.

$$CD = UD + PD$$

UD = *Cost of Unplanned Downtime*
PD = *Cost of Planned Downtime*

$$UD = \lambda \times MTTR \times T_U \times \text{Cost per Hour}$$

λ = *Steady-State Failure Rate (failures per hour)*

The steady-state failure rate of the system is the failure rate that is expected during the system's service life or it's useful life. This is the phase of life between early life (system burn in) and the wear-out phase of life when the system failure rate is expected to significantly increase and the system should be retired. During the service life phase of the system when the failure rate is considered to be steady state, this mathematical relationship can be used.

$$\lambda = \frac{1}{MTBF_{System}}$$

$MTBF_{System}$ = *Mean Time Between Failure of the System (hours)*

T_U = *Total Amount of Run Time of the System During the Service Life (hours)*

Run time for electronics usually includes the time that the system is powered on while doing its job and in an idle state.

MTTR = Mean Time to Repair (hours)

MTTR is not just the time to repair or replace a failed component. It includes the:

- Time to detect a failure
- Time to diagnose the system and understand which system component(s) failed
- Time to access and repair or replace the failed component(s) (having spares and/or redundancy will greatly impact this)
- Time to verify the system is repaired correctly
- Time to bring the system back online

It is easy to see that MTTR is very dependent on having spares, the system location, the design, and the type of failures that typically occur.

$$MTTR = \frac{\Sigma(\lambda_i t_i)}{\Sigma \lambda_i}$$

λ_i = *Failure Rate for the ith Failure Mode*

t_i = *Time to Repair the System After the ith Failure Mode Occurred*

The failure mode is defined as the type of failure that occurred or the root cause of the failure.

$$PD = (\lambda \times MPdMT + f_{PM} \times MPMT) \times T_U \times \text{Planned Downtime Cost per Hour}$$

The frequency that predictive maintenance should occur should correlate the to failure rate of the system. Instead of performing maintenance after a failure has occurred, maintenance is performed at a planned time after a potential failure condition is detected but before the failure occurs.

MPdMT = Mean Predictive Maintenance Time (hours)

MPdMT includes the:

- Time to access
- Time to service and or replace component(s) (having spares and/or redundancy will greatly impact this)
- Time to verify the system is operating correctly
- Time to bring the system back online

$$MPdMT = \frac{\sum(\lambda_i t_i)}{\sum \lambda_i}$$

λ_i = *Frequency of the ith Predictive Maintenance Activity*

t_i = *Time to Conduct the ith Predictive Maintenance Activity on the System*

f_{PM} = *Frequency of Preventive Maintenance (per hour)*

MPMT = Mean Preventive Maintenance Time (hours)

MPMT includes the:

- Time to access
- Time to service, replace, and/or calibrate component(s) (having spares and/or redundancy will greatly impact this)
- Time to verify the system is operating correctly
- Time to bring the system back online

$$MPMT = \frac{\sum(f_i t_i)}{\sum f_i}$$

f_i = *Frequency of the ith Preventive Maintenance Activity*

t_i = *Time to Conduct the ith Preventive Maintenance Activity on the System*

Cost of Maintenance (M)

$$M = PdM + PM + CM$$

PdM = Cost of Predictive Maintenance
PM = Cost of Preventive Maintenance
CM = Cost of Corrective Maintenance

Cost of Predictive Maintenance (PdM)

$$PdM = \lambda \times T_U \times PdM\ Event + Cost\ of\ Tools$$

PdM Event = Average Cost of a PdM Event

PdM Event = (MPdMT x Planned Downtime Labor Cost per Hour) + Service or Replacement + Spares + Logistic Cost

Planned downtime labor cost includes the cost of labor to perform predictive or preventive maintence for a system and the cost of training the labor force estimated on a per hour basis.

Cost of Tools = Cost of Software and Hardware Tools Needed for PdM

The cost of tools is typically a one-time expense that includes the cost of:

- Condition monitoring hardware and software
- Tools to remove and replace components
- Verification test equipment and software (which could be the same used for corrective maintenance)
- Maintenance of the equipment and software

NOTE: The tools can often be used for predictive, preventive, and corrective maintenance. If the tools can be used, then the cost of tools needs to be accounted for only one time and not for all three types of maintenance.

As shown above, MPdMT is greatly affected by having the right equipment/tools available and well-trained personnel.

Cost of Preventive Maintenance (PM)

$$PM = f_{PM} \times T_U \times PM\ Event + Cost\ of\ Tools$$

f_{PM} = *Frequency of Preventive Maintenance (per hour)*
PM Event = Average Cost of a PM Event
PM Event = (MPMT x Planned Downtime Labor Cost per Hour) +
Calibration, Service or Replacement + Spares + Logistic Cost

The smaller the MPMT of a system is, the lower the cost of predictive maintenance. As shown above, MPMT is greatly affected by having the right equipment/tools available, well-trained personnel, and a good calibration strategy. Many system suppliers can offer calibration options. Depending on the situation, it may be more cost-effective to have on-site calibration services or purchase a calibration service agreement from the system supplier. A standard supplier calibration program may be sufficient.

Cost of Corrective Maintenance (CM)

Failures can cause collateral damage to other equipment and/or the loss of the product that is under test when the failure occurs, causing the product to be scrapped. This cost is sometimes much costlier than the primary failure. For example, when an oil pump in a car fails, it can cause significant damage to the rest of the engine. Thus, a $100 oil pump can lead to a $5,000 engine rebuild.

$$CM = \lambda \times T_U \times CM\ Event + Cost\ of\ Tools$$

CM Event = Average Cost of a CM Event
CM Event = (MTTR x Unplanned Downtime Labor Cost per Hour) +
Repair or Replacement + Spares + Logistic Cost

Unplanned downtime labor cost includes the cost of labor to repair a system and the cost of training the labor force estimated on a per hour basis.

The smaller the MTTR of a system is, the greater the system availability and the lower the cost of unplanned downtime. As shown above, MTTR is greatly affected by location, the system design, having the right equipment/tools available, well-trained personnel, and a good sparing strategy. Many system suppliers can offer sparing options. Depending on the situation, it may be more cost-effective to own spares or purchase a service agreement from the system supplier to provide spares or have some hybrid agreement of the two. If the cost of unplanned downtime is low enough, on-site spares may not be justified and relying on standard supplier repair times may be sufficient.