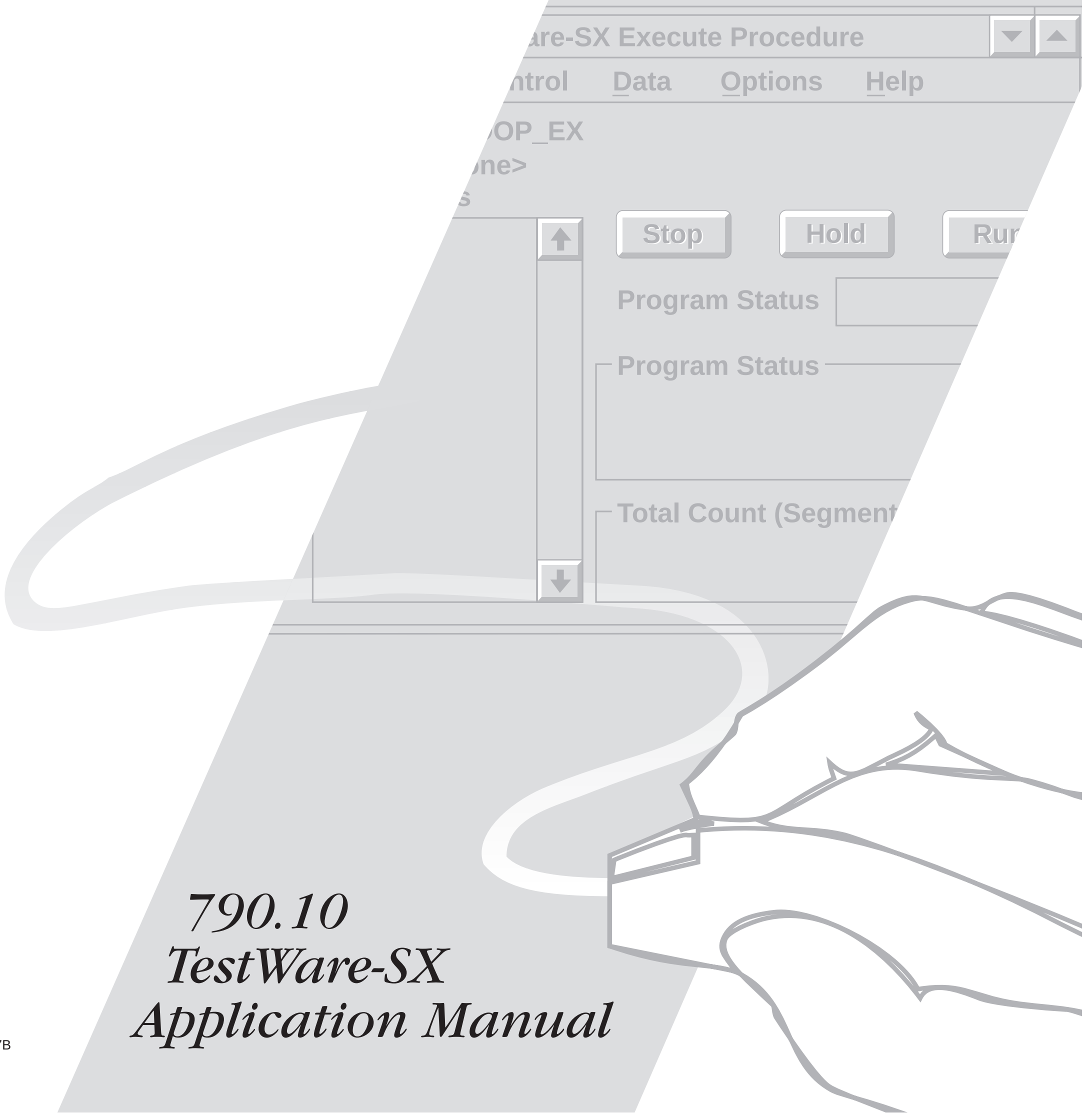
MTS®

TestStar™ II

Control System



790.10 TestWare-SX Application Manual

150197-07B

Proprietary data

Copyright information

Trademark information

Publication information

This manual supports the following TestWare-SX application release:

TestWare-SX Software Version	Manual Part Number	Manual Publication Date
Release 1.2A	150197-02A	September 1991
Release 1.3A	150197-02B	February 1992
Release 1.4A	150197-03A	June 1992
Release 1.4A	150197-03B	August 1992
Release 2.0A	150197-04A	December 1993
Release 2.0B	150197-04B	April 1994
Release 3.0A	150197-05A	January 1995
Release 3.1A	150197-06A	September 1995
Release 4.0A	150197-07A	November 1996
Release 4.0C	150197-07B	May 1997

Table of Contents

Preface 13

Chapter 1 Introduction 49

Chapter 2 Designing a Test 83

Chapter 3 Using TestWare-SX 135

Chapter 4 Template Windows 159

Chapter 5 Procedure Windows 197

Chapter 6 The Processes 227

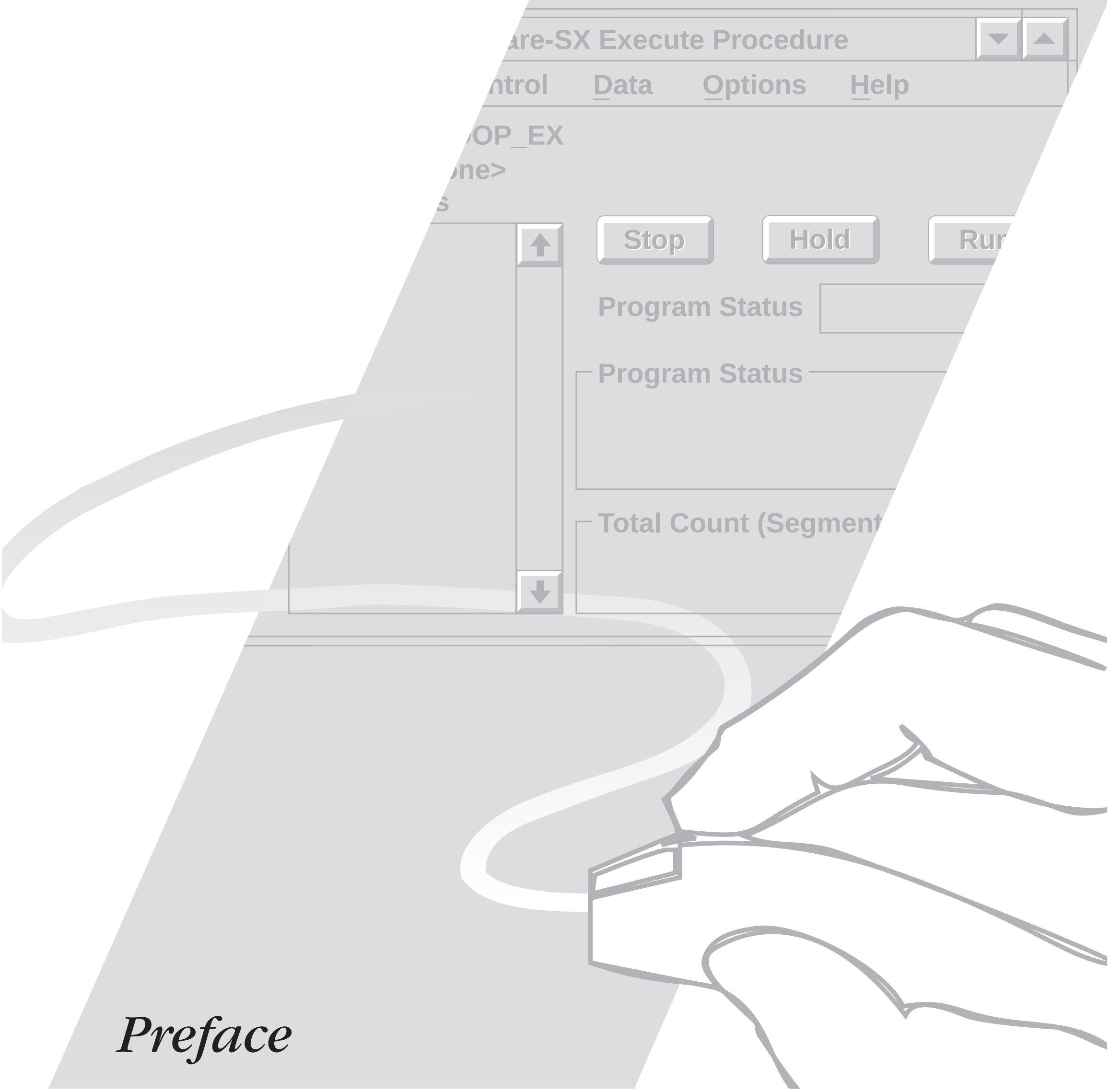
Appendix 358

Index 373

MTS®

TestStar™ II

Control System



Preface

Preface

What this manual does

This manual provides detailed information about TestWare-SX windows, menus, and processes. It is intended to serve as a continuing reference when you need to know detailed information about a specific subject. This manual also includes references to other chapters or manuals where related information may be found.

This manual also includes a chapter that shows you how to create a test template, edit a test template, and create a test procedure.

What this manual does not do

This manual does not provide detailed operating instructions to run a specific test. It is your responsibility to ensure that all test methods you use are accurate and safe.

What you need to know

TestStar is available for both IBM's OS/2 and Window's NT operating systems. You need to have a reasonable knowledge of your operating system before attempting to use this manual. You should know how to open and close windows, manage files, and so on.

- TestStar Version 3.1 requires Operating System/2™ (OS/2® Warp), Version 3.0 or higher.
- TestStar Version 4.0 and newer requires Windows NT™ Version 3.5.1.

Who should use this manual

This manual is designed for anyone who wants detailed information about any TestWare-SX window function or guidelines to design a test procedure.

Preface

Other Manuals

This manual is part of a set of TestStar manuals. The following describe the other TestStar and MTS system manuals.

- The **Reference Manual** (p/n 150194-xxx) describes every menu selection and how things work in every window for the TestStar application and all of the utility programs.
- The **TestStar Installation Manual** (p/n 150194-xxx) describes how to install TestStar and how to use the utility programs such as sensor calibration and system administration to establish the initial data base.
- **TestWare Application Manuals** describe specialized software for specific types of testing.
- The optional **C Programming Reference Manual** (p/n 150195-xxx) describes how to interface with TestStar using a high-level programming language.
- The **Product Information Manual** contains tabbed sections that describe the hardware components included with your system, such as your load unit and grips. This manual is primarily about hydro-mechanical products.
- The **Assembly Drawings Manual** contains tabbed sections that contain engineering drawings and part lists of many of the hardware components covered in the Product Information manual. This manual helps you to service your equipment and is useful for MTS Service Engineers if they service your equipment.
- The optional **TestStar A to Z** manual (p/n 150371-xxx) is an encyclopedia of testing. It describes testing terminology, concepts, and topics—from Actuators to Zeroing sensors.
- You may also have other manuals for components included with your system that are not manufactured by MTS, such as a printer manual or video monitor manual.

Installing TestWare-SX V3.1C

The TestWare-SX software is usually installed when the TestStar system software is installed. This is also true for software updates. This section describes how to install TestWare-SX as an add-on application.

Use this procedure to install the TestWare-SX software if it was not done during the TestStar software installation. The following procedure describes how to install TestWare version 3.1C. This version of TestStar is for OS/2 version 3.0 Warp.

Use the TestStar setup program to:

- Install the TestStar software for the first time
- Install TestStar software updates
- Redefine the TestStar hardware configuration
- Install any optional applications

Prerequisites

We assume that the OS/2 version 3.0 Warp operating system is properly installed in your computer. We also assume you have turned your computer on and have OS/2 running.

- You must install software updates in the order of release. For example, from version 1.4 to version 2.0x and from version 2.0x to version 3.0x.

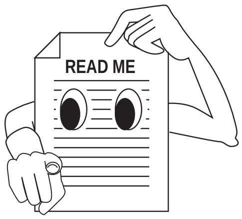
Before you begin

Locate the following floppy disks:

- The TestWare-SX Software disks.
- Read the README.TXT file (typically on disk 1) for late-breaking information that may not be included the manuals.

Preface

Read the readme file



MTS software typically includes a README.TXT file that contains late-breaking information not included in this manual. If the file is included, it should be opened and the information reviewed before installing the system software. The file is located on the disk.

Note *The software installation program will ask you if you have read the readme file. The installation program will automatically display the file if you want to see it.*

To read the file, insert the 790.XX application disk, double-click Drive A on the desktop then double-click the README.TXT file or open an OS/2 window and enter the following command:

TYPE A:README.TXT | MORE

Press any key to display the next page of the file. Press ^C (cntl + C) to exit the file.

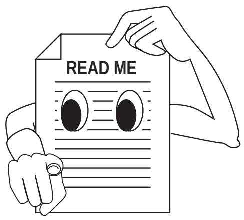
Procedure

The setup program has several operating modes. The following procedure describes only the Additional Application Installation mode. The Software Installation chapter of the TestStar Installation Manual (Chapter 4) describes how to use the setup program.

Step 1 Backup your disks

Make a copy of the TestWare-SX software. Use the copies to install the software onto the hard disk. Keep the original copy in a safe place.

Step 2 Read the readme file



MTS software typically includes a README.TXT file that contains late-breaking information not included in this manual. If the file is included, it should be opened and the information reviewed before installing the system software. The file is located on the disk.

Note *The software installation program will ask you if you have read the readme file. The installation program will automatically display the file if you want to see it.*

To read the file, insert the 790.XX application disk, double-click Drive A on the desktop then double-click the README.TXT file or open an OS/2 window and enter the following command:

TYPE A:README.TXT | MORE

Press any key to display the next page of the file. Press ^C (cntl + C) to exit the file.

Step 3 Start the software installation program

The TestStar setup program is called SETUP.EXE. This program is located in the TS2 directory of the hard disk after TestStar II is installed.

Note *The following assumes the setup program is in the default TestStar directory C:\TS2\SETUP.*

To start the setup program perform the following:

Double-click the OS/2 System icon
Double-click the Command Prompts folder
Double-click the OS/2 Window icon or the OS/2 Full Screen icon.
Type the command:

C:TS2\Setup

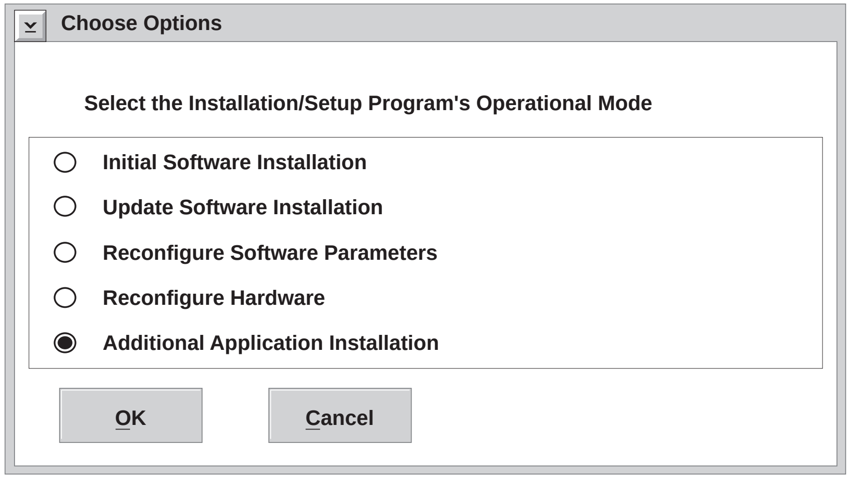
Preface

Step 4 Select the Setup Program's Operational Mode

The setup program displays the Ask Options prompt: Select **Additional Application Installation** to install additional applications. This mode selection is dedicated to installing optional application software.

Select Additional Application Installation from the list shown.

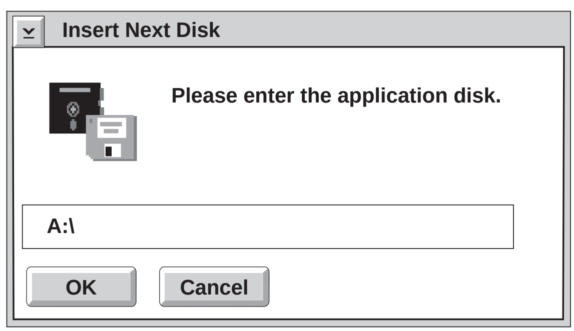
Press the OK button.



Step 5 Insert the application disk in the appropriate drive

Enter the drive location that you will use to install the software. Then install the application disk in that drive.

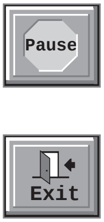
The default drive location is A.



Step 6 Press OK to start the software installation

Press the Pause pushbutton to suspend the installation

If you do not want to continue the installation, you can press the Exit pushbutton.

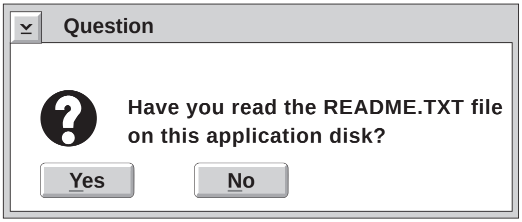


Step 7 Note about the readme file

If you select No, a message tells you that the readme file will be displayed. When you have finished reading the file, close the window to continue the setup program.

Select Yes to continue the installation procedure.

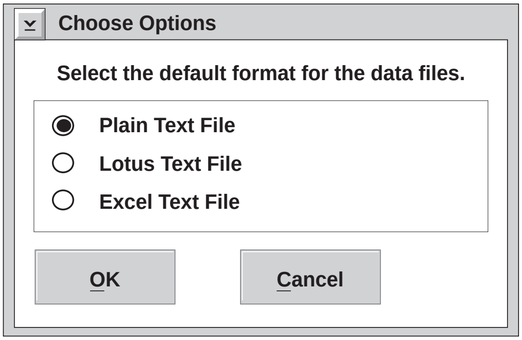
Select No to display the readme file.



Step 8 Select the data file format

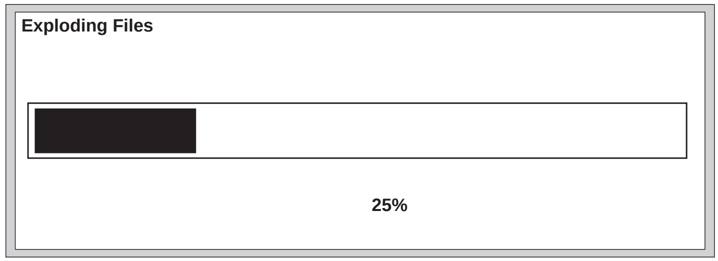
You can select one of the three file formats for the data acquisition process.

You may change the file format for any process with the Data File Options window.



Step 9 The setup program displays the installation progress

While the software is being installed, you will see the progress of the installation.

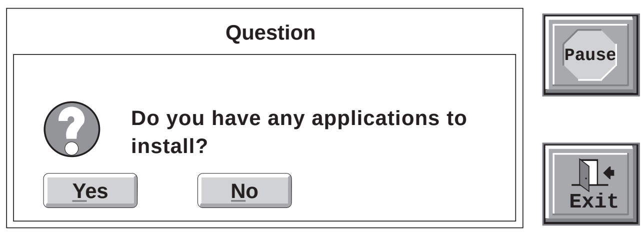


Step 10 Install any additional applications

Repeat steps 5 through 9 for each application you want to install.

Select Yes to install another application.

Select No if you do not want to install another application and end the setup program.



Step 11   Use the Administrator program to add the application

This step adds TestWare-SX to TestStar and allows you to select who can access it. If TestWare-SX is installed when TestStar is installed the program is automatically added to TestStar.

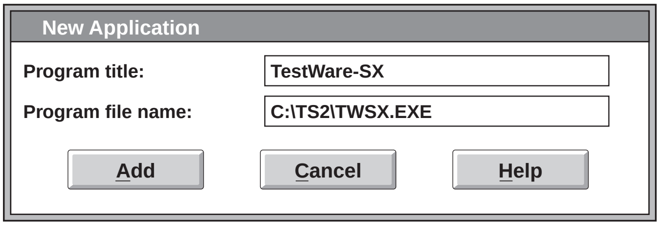
A From the OS/2 desktop, open the TestStar folder.

B Open the Utility folder.

C Open the Administrator program. This may cause the Login window to appear, login as needed. The System Administration window should appear.

D Select Applications in the Define menu, then press the Add pushbutton to display the New Applications window.

E Complete the New Application window as shown. Then press the Add pushbutton to display the User Access window.



F Highlight each of the users that can have access to the TestWare-SX application and press the OK pushbutton.

G When done, use the File menu to exit the program.

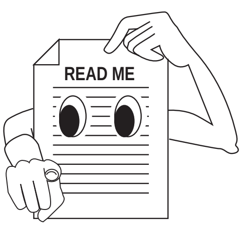
Installing TestWare-SX V4.0A

The TestWare-SX software is installed independently from the TestStar system software installation. This is also true for software updates. This section describes how to install TestWare-SX as an independent application. Use the following procedure to install the TestWare-SX software to a Windows NT operatin system.

Before you begin

- Locate the TestWare-SX System Software floppy disk.
- Backup your TestWare-SX templates and procedures. These files are located in the twsx directory (by default, TS2\TWSX).
- Read the README.TXT file for late-breaking information that may not be included the manuals.

Read the readme file



MTS software typically includes a README.TXT file that contains late-breaking information not included in this manual. If the file is included, it should be opened and the information reviewed before installing the system software. The file is located on the disk.

Note *The software installation program will ask you if you have read the readme file. The installation program will automatically display the file if you want to see it.*

A Insert the 790.XX application disk into DriveA

B Open the Main program group.

C Double-click the File Manager icon, or
double-click the Command Prompt icon

D If you opened the File Manager, click the drive A icon, then double-click the readme.txt program.

If you opened the Command Prompt window, type the command:

TYPE   A:README.TXT | MORE

Press any key to display the next page of the file.
Press ^C (cntl + C) to exit the file.

Preface

Prerequisites

We assume that the Windows NT version 3.51 operating system is properly installed in your computer. We also assume you have turned your computer on and have Windows NT running.

- You must be logged onto Windows NT as a user with administrator privileges.
- You must have Service Pack 4 or newer installed before you can install TestStar 4.0.

Abbreviated procedure

Step 1 Backup your disks

Make a copy of the TestWare-SX software. Use the copies to install the software onto the hard disk. Keep the original copy in a safe place.

Step 2 Start the software installation program

The TestStar setup program is called SETUP.EXE. This program is located the TestWare-SX installation disk.

To start the setup program perform the following:

A Insert the 790.XX application disk into DriveA

B Open the Main program group.

C Double-click the File Manager icon, or
double-click the Command Prompt icon

D If you opened the File Manager, click the drive A icon, then double-click the Setup.exe program.

If you opened the Command Prompt window, type the command:

A:\Setup

Step 3 Note about the readme file

The installation program asks if you have read the readme file.

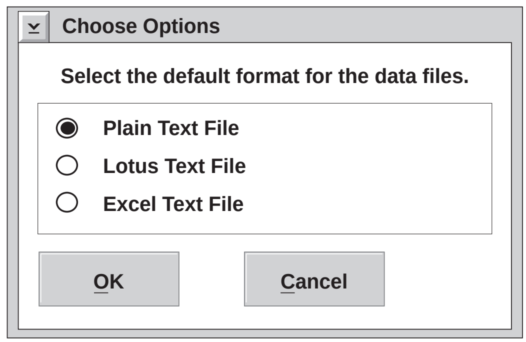
- Select **Yes** to continue the installation procedure.
- If you select **No**, the readme file will be displayed. When you have finished reading the file, close the window to continue the setup program.

Page 16 of this chapter has more information about the readme file.

Step 4 Select the data file format

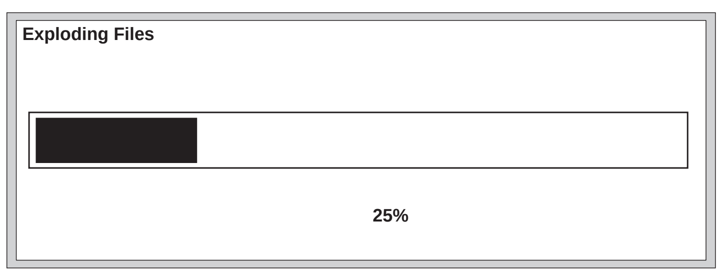
You can select one of the three file formats for the data acquisition process.

You may change the file format for any process with the Data File Options window.



Step 5 The setup program displays the installation progress

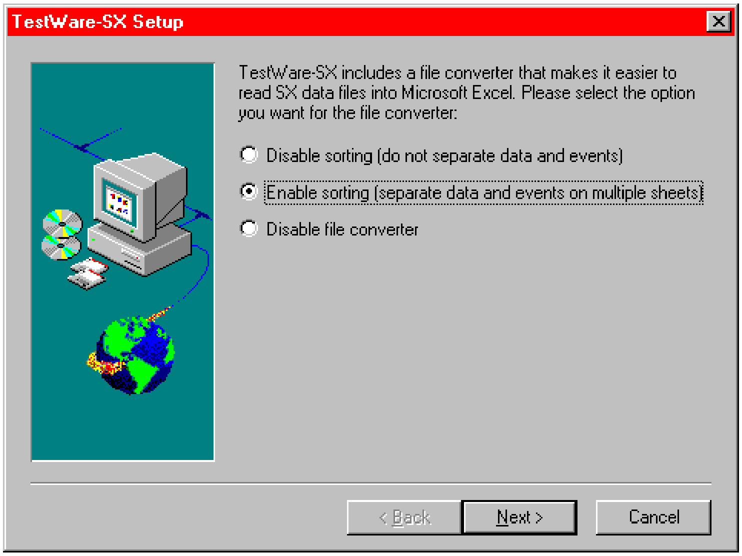
While the software is being installed, you will see the progress of the installation.



Step 6 Select the Excel file converter option

TestWare-SX includes a file converter for porting *.DAT files to Microsoft Excel. When selecting the Excel file converter option, keep the following in mind:

- Disable sorting – This option does not break the events, procedures, and data apart when converting files. It spreads the entire .DAT file over multiple sheets to prevent the Excel "too many lines" error.
- Enable sorting – This option puts events, procedures, and data on separate sheets. The three groups spread over multiple sheets to prevent the Excel "too many lines" error.
- Disable converter – This option disables the converter. You may rerun the install to load this converter at a later date.



Step 7 Use the Administrator program to add the application

This step adds TestWare-SX to TestStar and allows you to select who can access it. If TestWare-SX is installed when TestStar is installed the program is automatically added to TestStar.

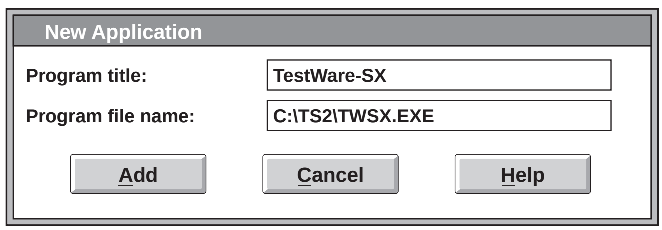
E From the desktop, open the TestStar folder.

F Open the Utility folder.

G Open the Administrator program. This may cause the Login window to appear, login as needed. The System Administration window should appear.

H Select Applications in the Define menu, then press the Add pushbutton to display the New Applications window.

I Complete the New Application window as shown. Then press the Add pushbutton to display the User Access window.



J Highlight each of the users that can have access to the TestWare-SX application and press the OK pushbutton.

K When done, use the File menu to exit the program.

Safety Precautions

⚠ WARNING

Improper system installation, operation, or maintenance can result in hazardous conditions that can cause severe personal injury or death, and damage to equipment or specimen.

Read these Safety Precautions before you use the equipment.

It is very important that you remain aware of hazards that apply to your test system. These Safety Precautions describe hazards that apply to your test system, and offer suggestions for avoiding hazards.

Overview

This chapter contains general operating safety techniques and precautions for operators of materials test systems.

Because each test system is configured for a unique application and operates within a unique environment, it is important to review these guidelines while considering your test system to ensure that the specific operating environment and operating procedures do not result in hazardous situations. Although complete elimination of hazards may not be possible, use the following guidelines to identify hazards so that appropriate training, operating procedures, and safety equipment can be set up.

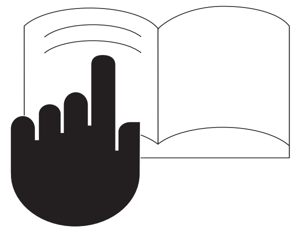
Common sense and a thorough knowledge of a specific system's operation and capabilities usually suggest the appropriate approach to system operation safety. Therefore, proper safety practices should begin with operator training. Operators should have had prior schooling and training on similar systems. (MTS has training classes that cover servo hydraulic operating theory, system operating procedures, and system maintenance techniques.) In addition, you should gain an understanding of system functions by studying the various instructions and manuals supplied with the test system.

General Safety Guidelines

The following safety guidelines are applicable to most test systems. As you read each item listed below, consider how it applies to your system. This can help to produce safer operating practices. If you have any questions, contact an MTS representative.

Know safety placards, read the manuals

Locate, read, and follow all instructions on equipment safety placards. Placard location is typically described in the installation section of the hydro mechanical product manuals.

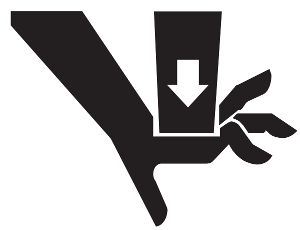


Know emergency stops

Know where all of the system **Emergency Stop** buttons are located so that you can stop the system quickly. **Emergency Stop** buttons have striping like the sample shown at the right.

Know potential crush points

Know where the potential load unit pinch and crush points are and take appropriate safety precautions. Refer to the discussion on crush point hazards.



Know system interlocks

System interlock devices should always be used and properly adjusted as described in this manual. Test all interlock devices for proper operation immediately before a test. Never rely on interlock devices to protect you. These devices are designed to minimize the chances of accidental damage to test specimens or to equipment.

Preface

Do not bypass the interlock chain

Do not use any interlock reset to bypass the interlock chain while attempting to start the hydraulic power supply. Doing this could cause the hydraulic pressure to be applied regardless of the interlock condition.

Do not disturb sensors

Do not bump, wiggle, adjust, disconnect, or otherwise disturb a sensor (e.g., an extensometer) when hydraulic pressure is applied and the system is operating under control from that sensor.

Ensure secure cable connections

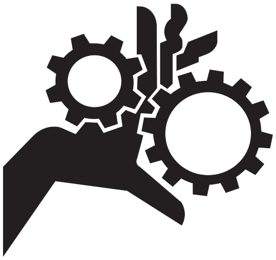
Do not change any cable connections with electrical power or hydraulic pressure applied. Changing cable connections with the system operating can result in an open control loop condition. An open control loop condition can cause rapid unexpected system response resulting in severe personal injury or death or damage to equipment. Also ensure all cables are connected if you make any changes in the system configuration.

Stay Alert

Avoid long periods of unvarying or monotonous work tasks that can contribute to accidents and hazardous situations. Familiarity with the working environment can lead you to overlook potential hazards in that environment.

Stay clear of moving equipment

Keep clear of moving mechanical linkages. Also stay clear of connecting cables and hoses that move along with the specimen or equipment. Objects may get tangled or dragged along with moving equipment. Serious injury can be inflicted by very high forces that can be produced. These forces could pinch, cut, or crush anything in the path of the moving equipment.

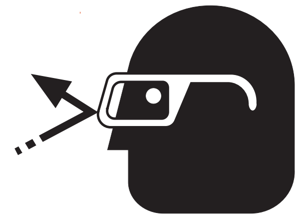


Be aware of electrical hazards

To minimize potential electrical shock hazards while the system electrical power is turned on, avoid touching exposed wiring or switch contacts.

Use eye protection

Use adequate eye protection when working with high-pressure hydraulic fluid or explosive specimens, and in circumstances during which anything peculiar to the specimen setup could break apart and cause eye injury.



Have first aid available

Accidents happen even to careful people. Arrange scheduling so that a properly trained person will be close by at all times to render first aid.

Practice good housekeeping

Keep work area floors clean. Hydraulic fluid spilled on any type of flooring results in a dangerous, slippery surface.

Keep bystanders away

Keep bystanders at a safe distance from all equipment. Never allow bystanders to touch specimens or equipment while the test is running.

Wear proper clothing

Do not wear neckties, shop aprons, loose clothing, or long hair that could get caught in equipment and create a potentially injurious situation.

Safety Guidelines to Follow While Operating the Equipment

Know proper system operation

Do not make mechanical or electrical adjustments to system components unless you know exactly how the adjustment will affect system operation. Consult your MTS representative when in doubt about any adjustment procedure.

Know results of using system controls

Do not make any unnecessary adjustments during operation of the system. To avoid erratic or unexpected system response, do not make any adjustments while the system is operating unless specifically instructed to do so.

Know crosshead lift and lock controls

Unlock the crosshead only with high hydraulic pressure applied. Do not adjust the lift controls when the crosshead is locked.

Know when to turn on hydraulics

Turn off hydraulic power except for those times that it is necessary for specimen setup or to run the test. Do not have hydraulic power on when making changes to the system configuration.

Know system control electronics

Have a thorough knowledge of the control electronics before turning on hydraulic power. Always follow the recommended operational procedures to turn on hydraulic power—failure to do so can cause the actuator to move rapidly and unexpectedly.

Know system hydraulic configuration

Some test sites have multiple test stations served by one hydraulic power supply. Understand how these units are interconnected before turning on hydraulic power.

Check system cabling

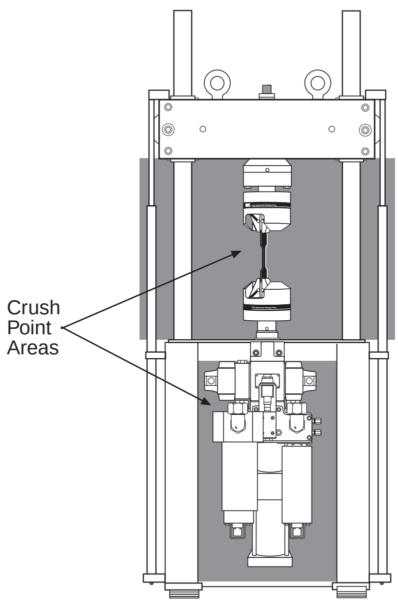
Check the cabling to the system sensors and servovalve. If the feedback or servovalve signal is lost for any reason (such as the connector coming loose or not connected, or the cable is damaged), the resulting signal loss will cause the actuator piston to move at maximum force and maximum velocity until it reaches a mechanical limit. Anything in its path (including you) could be crushed.

Make a trial run

Before operating the system for the first time, make a trial run through the desired test by locating the window controls involved without actually performing the adjustment or turning on hydraulic power.

Load Units and Other Crush Point Hazards

It is especially important to stay clear of any potential crush points when the system is operating. Know where the crush points are in your system and protect yourself and others from those crush points with appropriate safety devices. The following paragraphs describe crush points and precautions to take while working around crush points. These paragraphs apply to most testing and production systems.



- Keep clear of any mechanical linkage that moves within a closed area. If the linkage should move (when the system starts or due to mechanical failure), very high forces can be present that could pinch, cut, or crush anything in the path of linkage movement.
- Never allow any part of your body to enter the path of machine movement or to touch moving machinery, linkages, hoses, cables, specimens, etc. These present serious crush points or pinch points.
- A crush point exists between the platen and crosshead on load units where the actuator piston rod and specimen move. Another potential crush point exists where the lower end of the actuator piston rod extends below the platen and the bottom of the load unit/load frame.

Avoiding Hazardous Actuator Movement

The high forces and rapid motions that are usually present in testing systems can produce destructive forces from unexpected or uncontrolled actuator response.

CAUTION

Several things can cause unexpected actuator movement.

The possible combinations of system hardware and software settings make it impossible to predict conditions that produce unexpected actuator movement.

Some conditions can cause an actuator to slam to its mechanical limit, smashing anything in its path. Some conditions can cause an actuator to react so slowly to a command it may appear not to be working. And some conditions can cause the actuator dance in an unstable fashion while making an obnoxious noise.

Following are safety precautions for you to take:

- If the control mode feedback signal is interrupted during operation (e.g., if a sensor or servovalve cable is disconnected or breaks), the digital controller senses an error and causes the actuator to attempt to correct the error by stroking at maximum force and maximum velocity until it reaches an internal limit or external mechanical obstruction (e.g., tools, specimens, hands). The full force of the actuator will be applied to that limit or obstruction. (A selectable operating range does not reduce the force capability, it only increases the sensitivity of the electronic components.) To avoid a control mode feedback signal loss, protect sensor cables from damage, and never connect or disconnect any cable with electric of hydraulic pressure applied. If the feedback signal is lost, remove hydraulic pressure immediately. An open control loop also results if the cable from the digital controller to the servovalve is disconnected or broken while hydraulic pressure is applied.
- The composite command signal for the servo control loop may consist of several program inputs. If one of these inputs is suddenly changed while hydraulic pressure is applied, the servo control loop will sense a large instantaneous error and the actuator will respond accordingly. Do not make any program changes unless you know exactly how the change will affect operation.

- An unexpected actuator stroke or excessive actuator force can result from over programming. The composite command to the servo control loop is the algebraic sum of the Function Generator window's **Mean Level** and **Amplitude** inputs; either can program ±100% of the system's force-producing capability. For example, in most systems, a ±10 volt signal produces full system response: if the input to the controller is a ±10-volt sine wave and **Amplitude** is adjusted to the maximum setting, any mean level offset introduced by the **Amplitude** control causes the command to exceed the capabilities of the system. When determining program commands, make sure to avoid over programming.
- Many systems contain hydraulic accumulators that store enough energy to temporarily operate the actuator at full force capacity when the hydraulic pressure is shut off. For this reason, the usual interlock devices will not prevent hazardous actuator stroking.
- The failure or shutoff of electrical power to the testing system while hydraulic pressure is applied will cause considerable, unpredictable actuator reaction due to stored energy in the accumulators and irregular pump shutdown. Under these conditions, the actuator will generally stroke at maximum force and maximum velocity in either direction or, if a specimen is attached, apply full tensile or compressive force (i.e., positive or negative acceleration). Ensure that electrical power connections are not interrupted during test system operation.
- Do not use any interlock reset to bypass the interlock chain and attempt to start the hydraulic power supply. Doing this will cause the hydraulic power supply to start and hydraulic pressure will be applied regardless of the interlock condition. The error detector may be adjusted to trip whenever a large error is present, preventing the continued application of hydraulic pressure.

Guidelines For Installing Specimens

Because you are very close to or in contact with the system force train during specimen installation, this procedure can be the most hazardous part of system operation. Because it is usually necessary to have hydraulic power turned on, follow all of the instructions in this manual in addition to the following:

- Clear the work area, especially near system crush points.
- Ensure that the servo control loop is properly phased and stable (refer to the TestStar Installation Manual for procedures). Be particularly alert for phase or control reversal if the system setup has been modified since the previous operation. If operating the system in force or strain control, adjust the gain control to a value known by experience to be stable for the particular specimen in use.
- Use extreme caution when handling or supporting the specimen so that fingers and hands are never exposed to potential crush points during specimen installation. Use tongs to handle the specimen.
- To move the crosshead on load units not equipped with hydraulic lifts, support the crosshead using a lifting device capable of supporting the crosshead weight plus the weight of any fixtures and grips. Remove any slack from the crane cable or chain before unlocking the crosshead.
- A hazardous situation exists when air becomes trapped inside the lift cylinders on load units equipped with hydraulic lifts. Trapped air can cause erratic movement of the crosshead when the lift controls are operated. After installation, or if the crosshead does not move smoothly, bleed the lift cylinders as directed in the load unit product manual. Stay clear of the lower platen and the crosshead when operating the lift controls.

Checking the Hardware Setup

Always determine the necessary hardware configuration required for the test to be performed. Make all necessary changes to the configuration before applying electrical power or hydraulic pressure.

Check for hardware configuration changes

Due to the comprehensive nature of the system's testing capabilities, different types of tests may require changes in the hardware configuration to accommodate specific desired test results. Examples of hardware configuration changes include:

- Changing from one extensometer to another.
- Changing from a high-capacity force sensor to a low-capacity force sensor.
- Changing between servovalves on dual servovalve manifolds.

When you have multiple force sensors

If the system is configured to use more than one force sensor (e.g., typically, one with a force rating equal to system capability and another with a lower force rating), additional considerations may be necessary to protect the low capacity force sensor from damage. It does not reduce the full force capability of the hydraulic actuator. It only increases the sensitivity of the electronic control and readout components.

Installation and Modification Guidelines

The following installation and modification guidelines recommend design practices and modified system setup considerations that should be observed to minimize system operating hazards. Even when using the system for the first time and setup changes seem unlikely, a thorough understanding of the following guidelines will help in understanding system operation:

- Tests often operate for extended periods with no supervision and may attract spectators. This combination requires that any test laboratory setup provide adequate protection for bystanders as well as for system operators.
- Be sure to study the manuals to gain sufficient knowledge of system operation, and service and modification procedures.
- Refer to the TestStar Installation Manual for information about emergency stop connections on the digital controller rear panel.
- A competent engineer should be responsible for system installation or modification. The engineer must consider how changes to an existing facility or system might affect safety and reliability.

Supervising the System

The engineer responsible for any installation, modification or alteration to a test system should consider the following precautions:

- Protect all system hoses and cables from sharp or abrasive objects that could cause hose or cable failure. Route hoses and cables away from areas that expose them to possible damage.
- To avoid thrashing and subsequent deterioration, hydraulic pressure hoses should be anchored to the ground or tied to a corresponding return line within two feet of the flow outlet end. As an alternative, hoses can be run in trenches or other protected areas. Design enough fittings into a system to allow for the accommodation of dimensional errors without placing severe strains on the fittings or tube ends. Be sure to use appropriately rated fittings from a reputable manufacturer. Pressure line hoses for normal hydraulic service should have a burst pressure at least four times the operating pressure.
- Ensure all operators are familiar with any changes to the test system and provide training on how the changes affect operation and maintenance.

The Importance of Proper Maintenance

Proper maintenance is important to system operating safety. Without good maintenance practices, system reliability and safety degrades to the point where potential hazards can become extreme dangers. Study the manuals and the following paragraphs before beginning any type of system maintenance.

- Service must be done only by qualified persons.
- The service procedures in the individual product manuals are effective ways of maintaining the units. Read the procedures before you start working on a unit, then follow them carefully. In other words, don't get inventive.
- Use only designated MTS replacement parts. Parts not approved by MTS can adversely affect safety in addition to degrading reliability, increasing maintenance downtime, and voiding warranty coverage.
- Perform all calibration procedures in the TestStar Installation Manual to avoid improper signal scaling. Electronic signals between system components interact to operate the entire system.
- Systems that use pneumatic devices (e.g., accumulators, certain tandem and high-rate actuators) contain high-pressure gas that is very hazardous if improperly handled or poorly maintained. Read all gas cylinder labels to properly identify the type of accumulator and the type of gas used.
- Follow all accumulator charging instructions given in the manuals. When charging accumulators, use only dry nitrogen. (Dry nitrogen can be labeled "oil pumped" or "dry water pumped.") Do not use oxygen in place of nitrogen. If oxygen comes in contact with hydraulic fluid (e.g., if an accumulator bladder ruptures or leaks), a highly explosive condition will exist. When in doubt about any nitrogen charging procedure or about any type of accumulator, consult MTS Systems Corporation (refer to the Preface for information on technical assistance).
- Protect electrical cables from spilled hydraulic fluid and excessive temperatures that can cause cable hardening and can eventually result in cable failure. Clean spilled hydraulic fluid from cables as soon as possible.

- Inspect all cables for cuts, exposed wires, or other types of possible damage prior to system operation. Cable connectors must be securely plugged into their respective receptacles. Inspect each cable where it enters the cable connector for signs of excessive flexing (broken insulation) or exposed wires.
- Remove all system power before replacing any cable found to be defective. Ensure that all cables have appropriate strain relief devices installed at the cable and near the connector plug. Do not use the connector plug as a strain relief.
- Thoroughly inspect hoses for blisters, cuts, or other damage prior to system operation. Any weakening of the wire wrapping or reinforcing should be considered cause for hose replacement. While the system is operating, inspect all hoses and cables to ensure that there is no excessive thrashing, bending, or chafing that could cause cable or hose damage.
- Flush the hydraulic system immediately after any of the system's piping (i.e., hoses, hard lines, servovalve, hydraulic power supply components, etc.) has been replaced or its configuration has changed.
- Special safety considerations are necessary when operating a system that contains fire-resistant hydraulic fluid. These fluids are usually toxic and can present a lethal situation if fluid is accidentally swallowed or if a sufficient amount of fluid is absorbed through the skin. Avoid breathing the vapor or mist from these fluids, do not eat or smoke while working with these fluids, and practice absolute personal cleanliness when working with these fluids. Do not mix fire-resistant fluids with petroleum-based fluids. Also, do not add fire-resistant fluids to systems incompatible with these types of fluids (doing so will destroy seals and severely damage the equipment).

Hazard Conventions Used in This Manual

;The following techniques are used to highlight special types of information.

WARNING

Warnings alert you that something hazardous can occur if you do not follow the instructions carefully. Physical injury to you or to the machine (or both) will likely be severe.

The plain (unbolded) text below the initial bolded sentence gives you additional instructions about how to avoid the hazard.

CAUTION

Cautions alert you that something hazardous can occur if you do not follow the instructions carefully. However, the personal injury or equipment damage will likely be moderate.

Cautions are also used for procedures that can cause loss or corruption of computer programs or data.

NOTE

Notes are used to point out especially important information that you should know before performing an operation, but failure to do so is not likely to result in a hazard.

Boldface text

Boldface terms such as **Emergency Stop** are direct references to physical control and indicator labels on the test system.

How to Obtain Technical Assistance

If you have any questions about an MTS system or product, contact the MTS corporate service center.

Note *Review the following pages for information about what to expect when you contact us.*

Address

MTS Systems Corporation
Service Support Group
Technical Support Department
14000 Technology Drive
Eden Prairie, Minnesota 55344-2290

Telephone

In the United States (all 50 states)	HELPLine (800) 328-2255
Outside U.S.	Contact your local service center

Telex

29-0521

Fax

Technical support questions	(612) 937-4766
General questions	(612) 937-4515

Internet

E-mail	Info@mts.com
Internet Home Page	http://www.mts.com

Preface

What to Expect When You Call

Your call will be registered by a HELPLine agent. The agent will ask for you site number. If you do not have an MTS site number or do not know your site number, you should contact your MTS sales engineer.

The HELPLine agent may also ask to verify the following information:

- Your company's name
- Your company's address
- Your name and the telephone number where you can normally be reached.

If you have called before regarding this problem, we can recall your file. You'll need to tell us the following:

- The MTS work order number.
- The name of the person who helped you.
- Be prepared to respond to questions when interfacing with MTS technical support personnel. We may ask you to perform certain tasks so we can locate the source of the problem.

Before you call

Prepare the following information before you call HELPLine support to prepare for the troubleshooting process.

Know your site number and system number.

Describe the problem you are experiencing:

- How long has the problem been occurring?
- Can you reproduce the problem?
- Were any hardware changes made tot he application or system operating software before the problem started?

Have the following information available:

- If relevant, print-outs of configuration files, and test procedures.
- The type or model number of your test frame, load unit, etc.

- The type of model number of your controller
- Model number and size of your hydraulic service manifold
- Serial number of any suspect component

If you are experiencing a computer problem, please have the following information available:

- Manufacturer's name
- Manufacturer's model number
- Type of system memory
- Amount of system memory
- Floppy drive information (model number, size, and capacity)
- Hard drive information (model number, size, and capacity)
- Manufacturer of printer/plotter and model number
- Mouse information (bus, serial; connected to what port?)
- Graphics board information (manufacturer and model)
- What other boards are installed in the computer?
- Is the system part of a network?

If you are experiencing a software problem, please have the following information available:

- Operating software information
  - What type of operational software are you running?
  - What version level of operating system is running?
  - What window type is used?
- Application software information:
  - What applications are you running? (MultiPurpose TestWare, etc.)
  - Know the version of each software application involved.

Other software being used:

- What other software was running when the problem was encountered? This could include such things as screen savers, keyboard enhancers, print spoolers, etc.
- Know the name and version of each software program involved.

While on the phone

Prepare yourself for troubleshooting while on the phone:

- Try to call from a telephone close to the system so that you can conduct some active testing over the phone.
- Have the original operating and application software disks available.
- If you are not familiar with all aspects of the operation of the equipment, have the necessary people available to assist you.

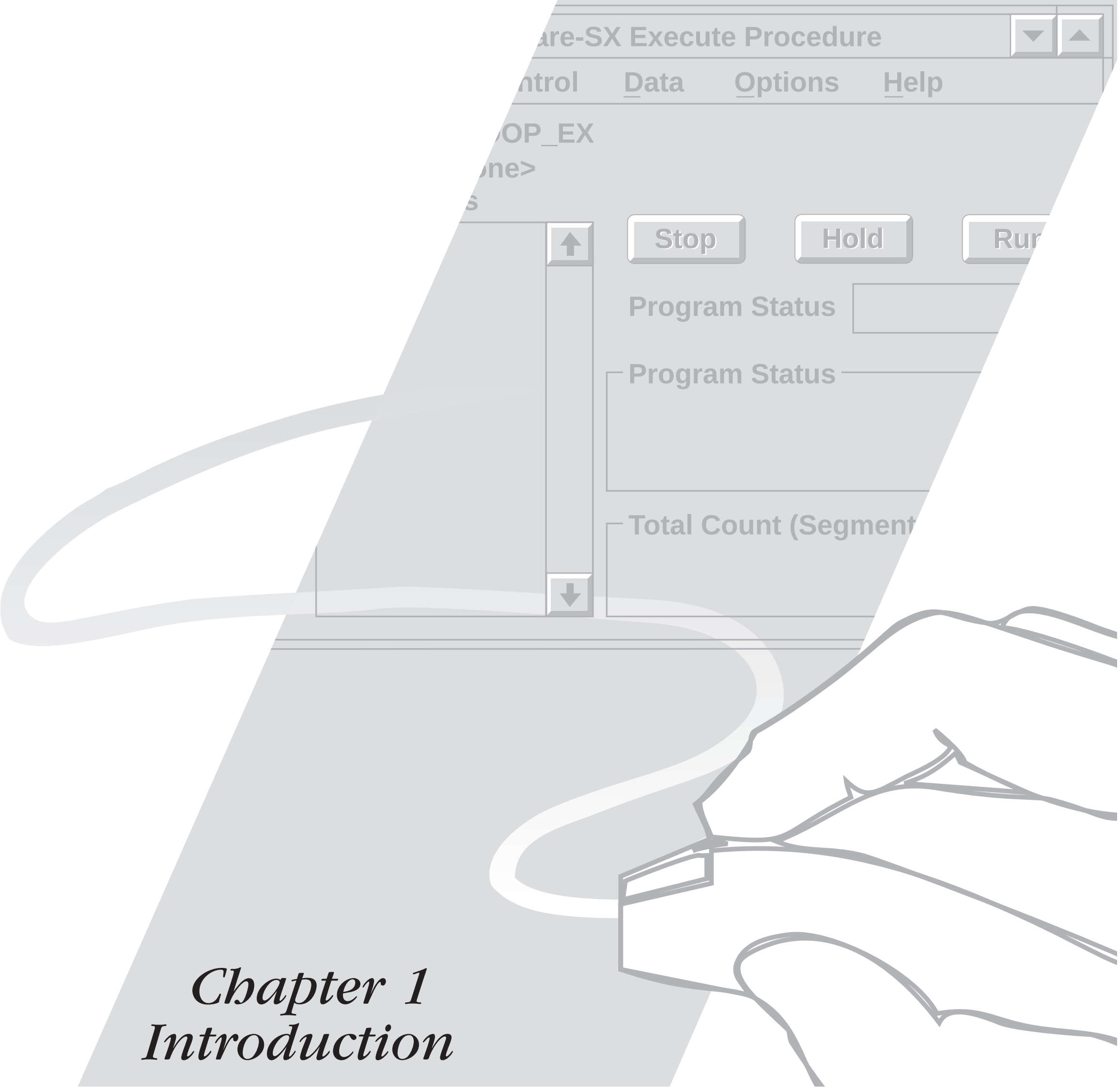
Prepare yourself in case a call back is required:

- Remember to ask for the work order number.
- Record the name of the person who helped you.
- Make sure you are able to write down any specific instructions to be followed, such as data recording or performance monitoring.

MTS®

TestStar™ II

Control System



Chapter 1
Introduction

A **Test Template** describes a type of test, such as a high-cycle fatigue. It defines the basic sequence of tasks that produce a test. The *Test Template* has default test parameters, such as rates, end levels, or frequencies.

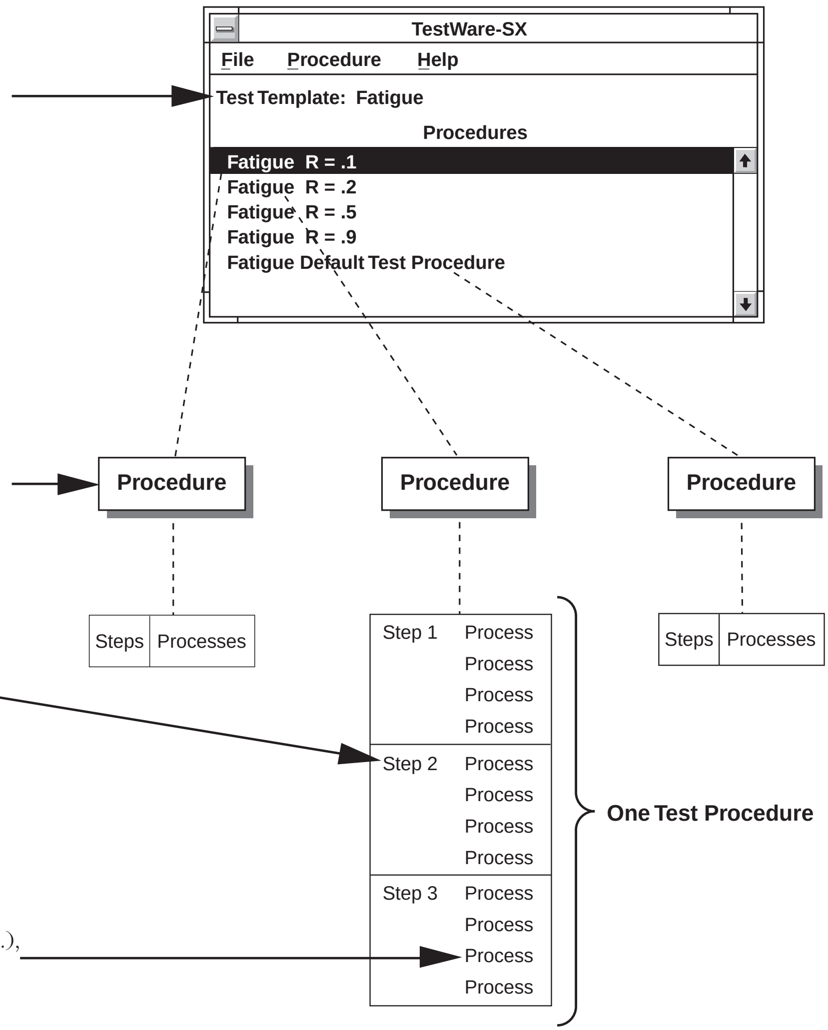
A **Test Procedure** is a specific test created from its parent *Test Template*, and has specific test parameters. One *Test Template* may list many different *Test Procedures*.

You select a *Test Procedure* to run a test.

A **Step** is a logical group of *Processes*. This modularizes a longer *Test Procedure* into smaller logical units.

A **Process** defines the actual activities during a test. These activities are:

- servoloop command signals (such as ramps or holds, etc.),
- control signals (to external devices), or
- data acquisition tasks.



Chapter 1

Introduction

Contents

Overview

Section A provides general information about this manual and an introduction to the terminology used by TestWare-SX.

Section B introduces the standard processes and provides information about optional processes.

Section C describes the basic concepts of designing a test template and defining a test procedure. This section also describes how to use triggers.

Section D provides information about the auto save feature and describes how to use it for test recovery operations.

Section A: General Information

⚠ WARNING

Read the Safety Precautions in this manual before you use the equipment.

The Safety Precautions portion describes hazards that apply to test systems and offers suggestions for avoiding hazards.

MTS test systems are powered by high-pressure hydraulic fluid. High-pressure hydraulic fluid is potentially dangerous. It is very important that you remain aware of hazards that apply to a test system.

What you need to know

TestWare-SX is available for both IBM OS/2 and Microsoft Windows NT operating systems. You need to have a reasonable knowledge of your operating system before attempting to use this manual. You should know how to open and close windows, manage files, and so on.

- TestWare-SX Version 3.1 use Operating System/2™ (OS/2®), Warp 3.0.
- TestWare-SX Version 4.0 and newer use Microsoft® Windows NT®.
- You also need to know how to use TestStar.

What this manual does

This manual provides detailed information about the TestWare-SX windows and menus. This manual also includes test design information and a basic operation procedure when using TestWare-SX to run a test.

What this manual does not do

This manual does not provide information about the basic TestStar windows or how to establish the required TestStar data base of files (configuration, sensor calibration, etc.). Refer to the TestStar manual set for further information.

Introduction

Other manuals

This manual is one of several manuals that may be included with your system.

- The **Reference Manual** (p/n 150194-xxx) describes every TestStar menu and window.
- The **Installation Manual** (p/n 150200-xxx) describes how to install TestStar, cabling information how to use the sensor calibration program, and how to perform the initial software settings to establish the initial TestStar configuration file.
- The optional **TestStar A to Z** manual is an encyclopedia of testing. It describes testing terminology, concepts, and topics—from Actuators to Zeroing sensors.
- The **Programming Reference Manual** describes how to interface with TestStar using a high-level programming language. Two programming manuals are available. One supports BASIC programming (p/n 150222-xxx) and the other supports the 'C' language (p/n 150195-xxx). Both manuals are available upon request.
- **TestWare Application Manuals** describe specialized software for specific types of testing.
- The **Product Information Manual** contains tabbed sections that describe the hardware components included with your system, such as your load unit and grips. This manual is primarily about hydro mechanical products. It does not include information about TestStar components.
- The **Assembly Drawings Manual** contains tabbed sections that contain engineering drawings and part lists of many of the hardware components covered in the Product Information manual. This manual helps you to service your equipment and is useful for MTS Service Engineers if they service your equipment.
- You may also have other manuals for components included with your system that are not manufactured by MTS, such as a printer manual or video monitor manual.

The Basic Windows

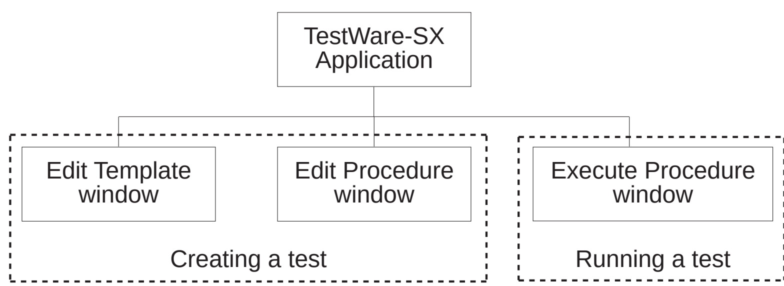
The TestWare-SX application is a flexible general-purpose program used to create and run a wide variety of test programs. TestWare-SX includes two major functions:

- **Creating a test** has two purposes. A template serves to establish the basic structure for a procedure. A procedure allows certain parameters to be changed for different iterations of the template.
- **Running a test** executes a specific test procedure.

Creating a test has two parts.

Creating a template produces a default procedure.

Creating a procedure produces different versions of the default procedure.



Terminology

The application uses several terms that you should become familiar with to understand its operation. The diagram below defines the main terms.

A **Test Template** describes a type of test, such as a high-cycle fatigue. It defines the basic sequence of tasks that produce a test. The *Test Template* has default test parameters, such as rates, end levels, or frequencies.

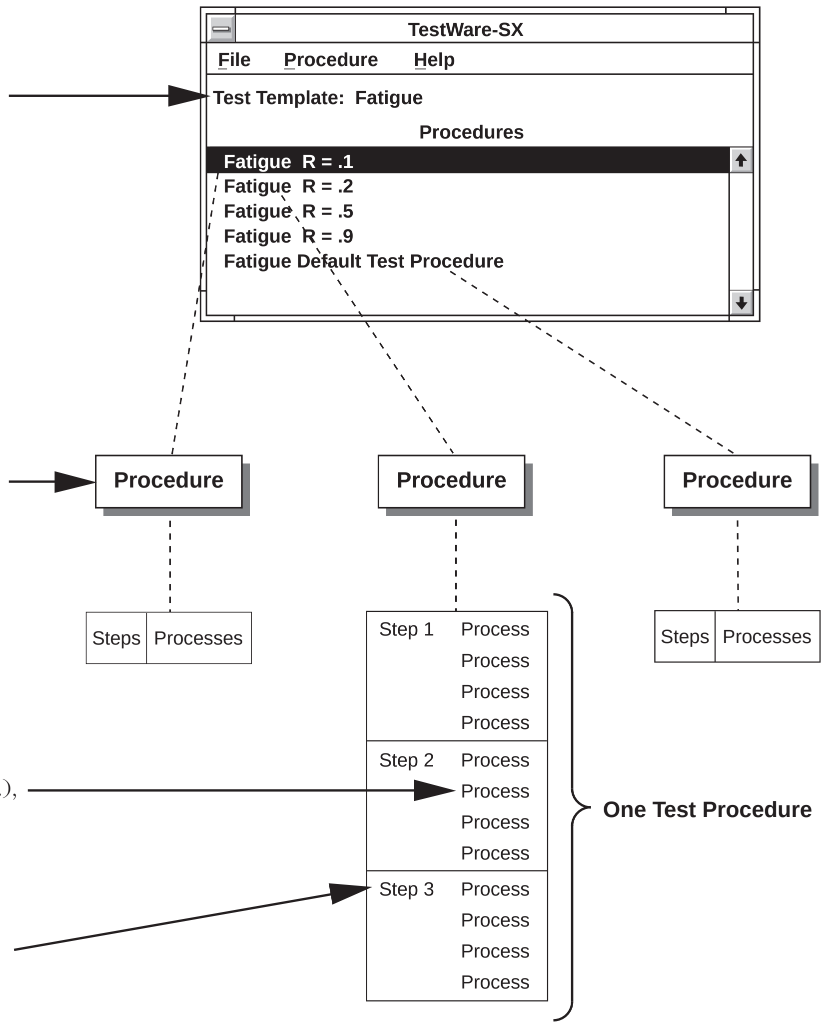
A **Test Procedure** is a specific test created from its parent *Test Template*, and has specific test parameters. One *Test Template* may list many different *Test Procedures*.

You select a *Test Procedure* to run a test.

A **Process** defines the actual activities during a test. These activities are:

- servoloop command signals (such as ramps or holds, etc.),
- control signals (to external devices), or
- data acquisition tasks.

A **Step** is a logical group of *Processes*. This modularizes a longer *Test Procedure* into smaller logical units.



Introduction

Section B: Processes

The process is the smallest unit in a test. Processes are the basic tasks that control the machine and take the data for the test. Each process has unique characteristics and capabilities.

Types of processes

TestWare-SX uses four types of processes. Each process is classified into the following categories.

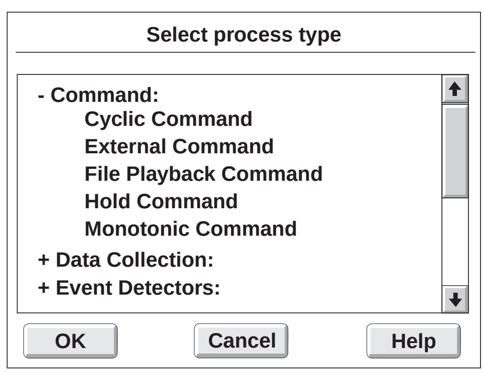
- **Command** processes control a servovalve or servo motor using a closed-loop control system.
- **Data Collection** processes accumulate raw sensor data.
- **Event Detectors** processes either respond to detectors of conditions or define conditions that trigger other processes.
- **External Control** processes issue signals to devices external to the servo loop control system.
- **Special** processes can combine command, data collection, event detectors, and additional capabilities into one process.

Selecting a process

Each process has unique characteristics and capabilities. You select the process type by using the window shown below. The processes are listed by category.

Highlighting a process (or category) and pressing the Help button displays information about the process.

Each category can be expanded (+) to list the processes or collapsed (-) to show the category name only.



Standard processes

The following table shows how the standard set of processes is categorized. If you have other processes listed, see to *Optional processes* for an explanation.

PROCESS	TYPE
Analog Output	External Control
Cyclic	Command
Data Acquisition	Data Collection
Data Limit Detector	Event Detector
Digital Input Detector	Event Detector
Digital Output	External Control
External Command	Command
File Playback	Command
Hold	Command
Monotonic	Command
Operator Event	Event Detector
Operator Information	Special
Peak/Valley Change Detector	Event Detector
Program Control	Special
Temperature Control	External Control
Temperature Data Acquisition	Data Collection

Optional processes and templates

You may have optional TestWare applications. There are two types of optional applications; those that are additional processes and those that are predefined templates.

Additional processes are listed in the appropriate category of the Select process type window. Use the optional processes like any other process when designing a test.

Predefined templates are tests that comply with industry standards. With predefined templates, you need only to assign parameter values. Double-clicking a predefined template starts TestWare-SX in the Edit Procedure mode. See Optional Processes and Templates in this section for a description of each option.

Command Processes

Command processes control a servovalve or servo motor using a closed loop control system. These processes must be sequenced in series to command a control channel. A single command process can be applied to more than one control channel.

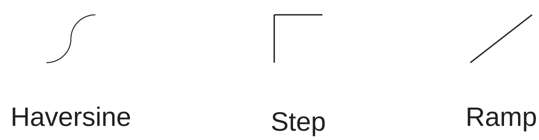
For example, a sequence consisting of a monotonic command followed by a cyclic command can control the servovalve of an axial channel.

Note *You should always have a command process in control of each control channel. If a gap in the command sequence occurs, the actuator position is maintained using the last control mode.*

Monotonic command

(*page 314*)

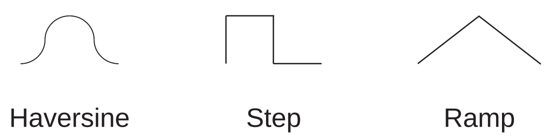
Monotonic command processes produce a single segment that starts at the present level and ends at a different level.



Cyclic command

(*page 240*)

Cyclic command processes repeat a waveform by assembling two monotonic segments and cycling them between two end levels. These commands can run continuously or for a predefined number of cycles.



External command

(*page 277*)

An external command process uses a signal from an external device (such as a function generator) to control the servovalve or servo motor. You can connect other command producing equipment to the TestStar rear panel and use it to run a test.

Hold command

(*page 310*)

Hold command processes hold the existing command for a specified time in a selected control mode.

File playback command

(*page 284*)

File playback command processes use a data file to define a series of monotonic segments. Each segment contains information that defines a waveshape, a rate type, and an end level. In other words, you can create a file that combines monotonic, cyclic, and hold processes.

This type of command is typically used to simulate spectrum or random waveshape sequences. The number of points in a file is restricted only by the capacity of the disk drive.

Multiple channels

Command processes can be applied to more than one control channel. In this case, some of the process parameters are shared between the control channels while others may be unique.

For example, a single monotonic command can be applied to multiple channels. The command applies the same waveshape and rate type to each control channel, but the control modes and end levels can be different for each control channel.

Data Collection Processes

A data collection process can acquire data from any sensor input signal. Data is stored temporally until is saved to disk in a data file. These types of processes not only collects data, but they are useful to trigger other processes.

Data acquisition

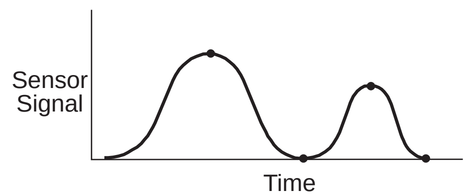
(*page 250*)

A data collection process acquires data from one master input signal and any of the other input signals. You configure the master channel to acquire data according to one of three modes of operation.

Data is temporarily saved in a buffer. The combination of the buffer size and type offer several options to acquire specific data and trigger other processes.

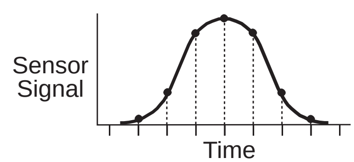
Peak/Valley and Valley/Peak Data

Records data when the master channel signal detects a peak or valley. The data mode can start with a peak or a valley.



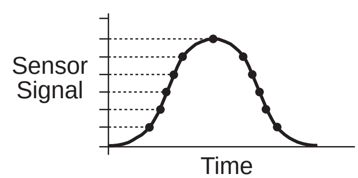
Timed Data

Records data at specified equal time intervals. Time becomes the master channel.



Level Crossing Data

Records data when the master input signal changes a specific amount.



Introduction

Temperature data acquisition

(*page 349*)

This process is like the data acquisition process except it uses only the timed data mode and a slow data acquisition rate (2 seconds). This process can acquire temperature data via an RS-232 interface (with MTS Series 409 Temperature Controllers).

Buffers

Each data acquisition process requires a buffer type and buffer size. These attributes determine how much data is collected and how the process affects other processes. The combination of the buffer type and buffer size can be used two ways: to acquire data and to trigger other processes.

Single buffer

Data is recorded to fill the buffer once, then stops the process and saves the data to disk. The size of the data buffer determines how much data to collect.

Use this buffer to collect specific data or trigger another process when specific data is collected.

Continuous buffer

Continuously records data to the buffer. When the buffer is full, the data is stored to disk.

Use this buffer to collect data throughout a test.

Continuous with trigger buffer

This buffer works the same as the *continuous* buffer except it also issues a trigger each time the buffer is full. This type of buffer ends at the end of the step.

Use this buffer to collect data and periodically trigger other processes.

Trigger-only buffer

This buffer type functions the same as the *continuous with trigger* buffer except it does not save data to disk.

Use this buffer to trigger other processes on a periodic basis.

Circular buffer

Continuously records data to the buffer. When the buffer is full, the newest data overwrites the oldest data.

Use this buffer to collect data near the end of a test.

Using data collection processes

There are three kinds of data that can be acquired and five kinds of buffers. The combination of the data mode and the buffer type provide a variety of ways to collect and use data.

- Single or multiple data collection processes can run simultaneously with the commands.

  For example, the test could be defined so that peak/valley data is taken simultaneously with timed data.

- A data collection process can start or stop any other process.

- Data collection processes are usually not processed sequentially with command processes.

  Reason: Useful data cannot be acquired without a command process causing the servo hydraulic system to do something. Data processes run in parallel with command processes.

- When creating a test where a command process is planned to start simultaneously with a data collection process, sequence the processes within the step so that the data collection process precedes the command.

  Reason: Because of timing considerations within the computer, there will be a slight delay between starting the two processes (even though they're both supposed to happen at the same time). If the command process precedes the data process within the template, there may be a slight loss of desired data collection at the leading edge of the physical motion.

Data Files

A data file contains the data acquired from a test along with a label indicating the type of data and the units of the data. Data can be acquired for all available input signals. The data file format can be selected for use with popular spreadsheet programs or plain text.

Below is an example of the output from a data file.

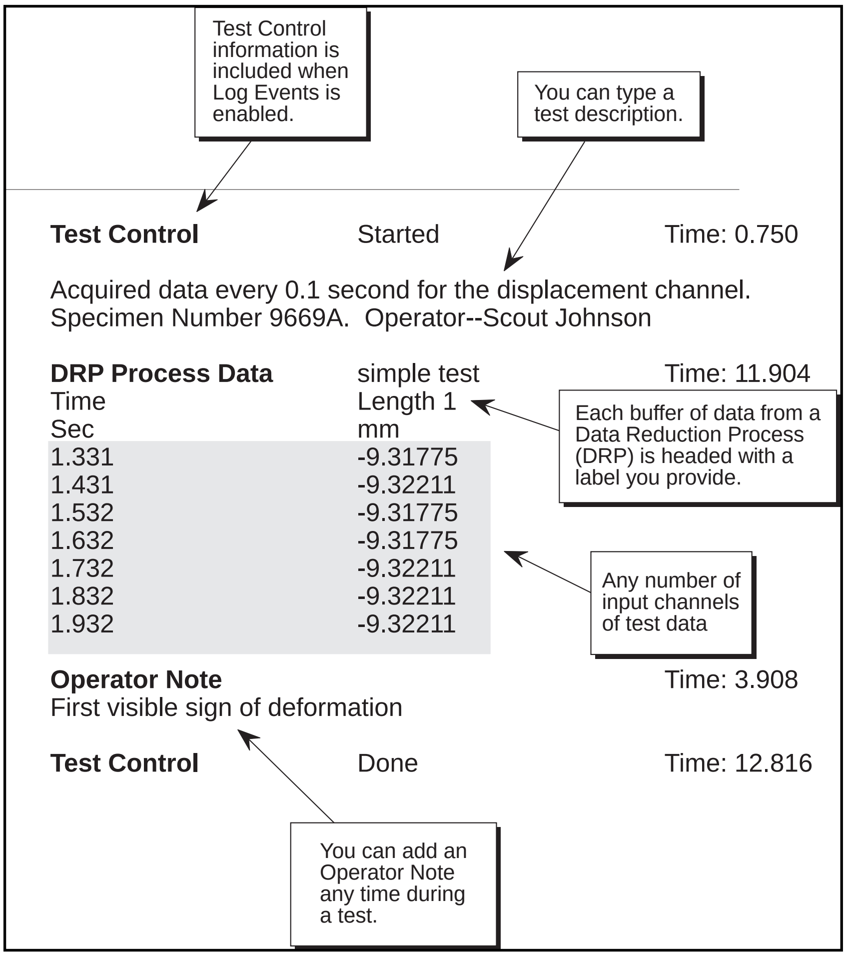
Data File Format

This shows the types of information that can be saved in a data file.

The data is output ASCII text that can be formatted for:

- Plain Text
- Lotus
- Excel

The file format contains tab separators so that the columns of data automatically appear in separate columns in a spreadsheet program.



Event Detector Processes

These processes either respond to detectors of conditions or create conditions that trigger other processes. Event processes are very effective at triggering other processes when specific conditions occur.

Data limit detector

(*page 261*)

This process monitors an input signal (sensor signal) to end one process and start another process. It can monitor:

- segment counts
- amount of time
- a sensor signal

Digital input detector

(*page 267*)

This process monitors up to 8 digital inputs from external sources to end one process and start another process. External signals can be input through the rear panel connector J54. This detector can monitor:

- high/low, low/high, or either transition
- channel low and channel high status

Operator event

(*page 320*)

An operator event process produces a button (and description) in a window. The button is usually set up to trigger another process. Up to three operator event processes can be active at one time. Each process can be configured to operate once or repeatedly.

This process allows you to manually interact with the test procedure. *For example*, a button could be set up to start a test after you perform some required preliminary action. Another button could be set up to record data each time you press it.

Peak/valley change detector

(*page 332*)

This process lets you monitor an input signal for changes in the peaks and valleys. The process begins by detecting a peak and valley, these values become the reference levels for the tolerance range. When a peak or valley exceeds a tolerance range the process triggers.

External Control Processes

Definition

External control processes issue control signals to devices external to the servo loop control system. Use the external control processes for the following:

- output signal status information
- trigger external devices to do something
- issue a voltage level

Analog output

(page 234)

The analog output process produces a voltage that can be output through one of the rear panel Readout connectors (J71 through J76). This process is useful for applications where a remote control voltage is needed.

Digital output

(page 272)

This process outputs up to eight 24-volt signals to external devices. The signals are available through the rear panel connector J55. Each output can independently do the following:

- set the output channel on (logic high)
- set the output channel off (logic low)
- toggle the output from its current state (low to high, or high to low)
- pulse the output by inverting the current output for a specified pulse width

Temperature control

(page 343)

This process communicates a temperature setting to an external temperature controller via an RS-232 interface. The temperature controller communicates the current temperature to the process. The process ensures that the temperature setting is maintained for a specific time before the process ends. This process is specifically designed for use with MTS Series 409 Temperature Controllers.

Special Processes

Special processes provide capabilities beyond the other category definitions. This type of process can combine the functions of the other categories along with additional capabilities into one process.

Most special processes are part of optional applications. Special processes can be designed for a specific type of testing. Special processes could be a specific combination of standard processes for a given test.

Program control

(*page 337*)

This process works like a custom interlock. Use this process to stop a test before it is complete. It is commonly used in conjunction with an event detector that triggers the program control process.

The program control process can do one of the following:

- display a message
- put the program in the hold status
- put the program in the stop status
- activate a hydraulic interlock
- generate an error

Operator information

(*page 326*)

This process lets you create a custom form that prompts an operator to input information. The information is added to the test data file. This process is useful for quality control applications where test related information (such as operator ID, batch number, part number, etc.) is included in the data file.

Optional Processes and Templates

The following are descriptions of the optional processes and templates.

790.13 Run-Time Plotting

This is a special process that provides on-screen plotting of selected channel data while a test is running.

790.14 Advanced Function Generation

Includes three command processes. The UDA Cyclic process allows you to define a segment shape that can be repeated. Mixed Mode Sine and Mixed Mode Pulse processes use two control modes – one as a mean level offset, another for the segment shape (sine wave or pulse).

790.15 RPC Utilities for TestStar

This process is similar to the standard File Playback process except that it is designed to play only RPC™ files. This process supports the RPC II™, RPC III™, and Component RPC™ formats, which are specific file formats. It also supports RPC III response data acquisition when an RPC III drive file is played.

790.16 High Speed Data Acquisition

This data collection process acquires data at rates up to 50 kHz (vs. 5 kHz for the standard data acquisition process). The high speed process acquires data according to a time increment using a single buffer. No other configurations are supported. This process also requires a second Model 490.40 Analog I/O module installed in the digital controller.

790.17 Data Monitor Processes

Includes two special processes designed for use with cyclic processes and the Dynamic Characterization process. Both processes plot signals periodically during long tests. The trend monitor process plots up to three sensor signals. The dynamic property process monitors up to seven calculated properties from the Dynamic Characterization process.

790.19 Run-time Processes

This includes two run-time processes. The Run-time Ramp Control process lets you simultaneously ramp up to four control channels in an incremental fashion, stepping toward indeterminate end levels.

The Run-time Plotting process displays a window while the process is running. The window shows up to three input signals. The example test template uses the run-time ramp control and plotting processes (with other processes) to create a static deflection test.

790.31 Dynamic Characterization

This process characterizes the dynamic properties of elastomeric materials and components. It allows you to sweep temperature, frequency, dynamic amplitude, mean level, and the phase relationship between control channels on up to four control channels simultaneously. It also allows you to nest sweeps inside one another. It also includes a set of macros for the Excel spreadsheet application.

790.33 Static Deflection

This process loads the specimen between predefined load or displacement end-levels and acquires subsequent timed data. Then it calculates the sample's stiffness as a cord or tangent modulus during the loading or unloading portion of the test, or as an average of the two. Finally, the process compares the calculated stiffness to predefined limits. It can also compare individual load and displacement data pairs to predefined envelopes for quality control applications. It also includes a set of macros for the Excel spreadsheet application.

790.35 Production QC

This is an optional feature for the 790.31 Dynamic Characterization process and the 790.33 Static deflection process. It is a hardware and software feature designed for the manufacturing environment. It includes a light curtain and control panel for a load unit. The light curtain allows you to install or remove a specimen when appropriate and shuts down the system if an object breaks the light curtain during a test. The Production QC feature can test different types of specimens (each with the appropriate test) and display the results (pass or fail).

790.37 Resonant Search

This process finds the frequency at which a specimen resonates. It is designed to describe the dynamic behavior of elastomeric materials and components in a free-end resonant mode. It lets you define one or more frequency sweeps at a constant amplitude. You can define each sweep to be linear or logarithmic.

790.38 Elastomer Tearing Energy

This process allows you to characterize the crack growth behavior of elastomeric materials. It allows you to define a loading schedule to acquire stress/strain data at each strain level of interest. When you start the test, the process allows you to enter crack growth data as it occurs, and allows you to modify the schedule to achieve the desired rate of crack growth. If desired, the process can also determine the pretest strain energy levels of the specimen.

790.61 Uniaxial Rock Mechanics

This application includes predefined templates for the following tests:

- Uniaxial Compression; ISRM "Suggested Methods for determining the Uniaxial Compressive Strength and Deformability of Rock Materials, ASTM D2938-86
- Uniaxial Compressive Deformability; ISRM, "Suggested Methods for determining the Uniaxial Compressive Strength and Deformability of Rock Materials", Part 2, ASTM D3148-86
- Uniaxial Direct Tension; ISRM "Suggested Methods for Determining Tensile Strength of Rock Materials", Part 1, ASTM D2936-84
- Uniaxial Indirect Tension; ISRM "Suggested Methods for Determining Tensile Strength of Rock Materials", Part 2, ASTM D3967-86
- Uniaxial Compress Deform Post Fail; ISRM, "Suggested Methods for determining the Uniaxial Compressive Strength and Deformability of Rock Materials", Part 2, ASTM D3148-86
- Uniaxial Creep, ASTM D43484

790.62 Triaxial Rock Mechanics

This application includes predefined templates for the following tests:

- Triaxial Compressive Strength, ASTM D2664-86
- Triaxial Creep, ASTM D4406-84
- Triaxial Compression - Single; ISRM "Suggested Methods for Determining the Strength of Rock Materials in Triaxial Compression"
- Triaxial Compression - Multiple; ISRM "Suggested Methods for Determining the Strength of Rock Materials in Triaxial Compression"
- Triaxial Compression - Continuous; ISRM "Suggested Methods for Determining the Strength of Rock Materials in Triaxial Compression"

790.63 Fracture Toughness for Rock

This application includes predefined templates for the following tests:

Fracture Toughness - Level I", ISRM "Suggested Methods for Determining the Fracture Toughness of Rock"

Fracture Toughness - Level II", ISRM "Suggested Methods for Determining the Fracture Toughness of Rock"

790.80 Resonance Control

This process finds the frequency at which a specimen resonates. It is similar to the 790.37 Resonant Search process except it is designed for structural components. It lets you define a frequency sweep at a constant amplitude. When the resonant frequency is found, you can specify a schedule to exercise specimen.

Section C: Test Design

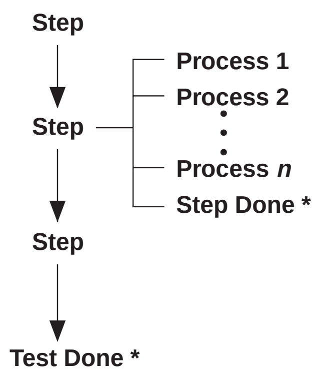
The main activities of designing a test

Activities 1 and 2 create a test template.

Designing a test includes four main activities:

1. Select an existing template or open a new template.
2. Create or edit a series of steps. Each step typically has at least two processes:

- A function generation command.
- A Step Done process.



* Each step includes a Step Done process. This process cannot be deleted.

Each template includes a Test Done step. You cannot edit or delete this step.

Activities 3 and 4 create a test procedure.

3. For each process, specify the specific test parameters: frequency, waveform shape, or data acquisition rate.
4. Name the test and save it. This is now a test procedure.

Look at an example

TestWare-SX includes some test templates (default procedures) and some test procedures that you may use or modify. You may want to open one up and look around.

Note *Appendix A includes a listing of each template and procedure. This is the same information you can obtain by using the print selections in the File menu of the Edit Template or Edit Procedure window.*

1. Open the Applications group in the main TestStar window.
2. Open the TestWare-SX application in the Applications Group window.
3. Select Open Template in File menu.
4. Open any of the templates listed.
5. Highlight the Default Procedure and select Edit Template in the File menu.

   At this point you can double-click a step and each process in the step to review how the triggers are sequenced.

Note *You can double-click the left mouse button on a process to display the parameters window. You can double-click the right mouse button on a process to display the design window.*

6. Double-click one of the procedures listed (including the Default Procedure).

At this point you can double-click a step and each process in the step to see how they are defined.

Creating a Test Template

Creating a template

A test template establishes the sequence of steps, the sequence of processes, and the types of processes.

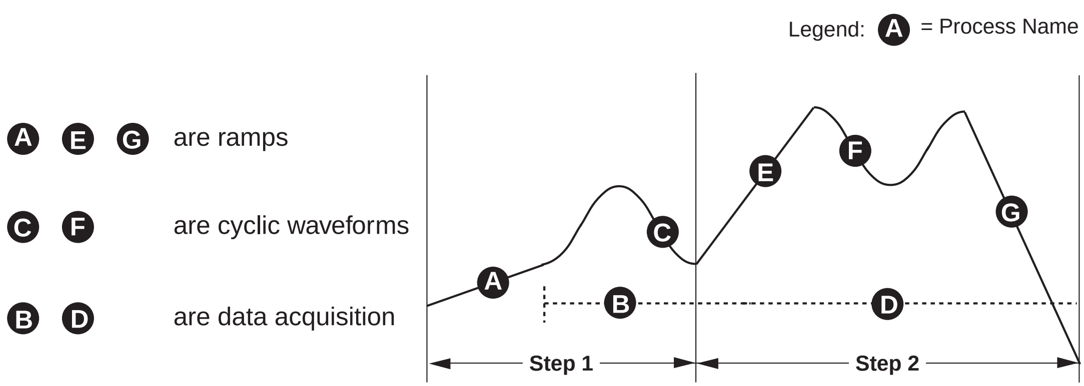
Each process requires the following information defined before the process can become part of a template:

- A process label (a name you assign to it).
- A control channel (for command processes).
- A start requirement, called the *start trigger.* A typical start trigger is the beginning of a step or the completion of another process.
- A stop requirement, called the *end trigger.* A typical end trigger is the end of the step or the completion of a process.

A sample template

The diagram below illustrates a simple template of two steps. Each step contains command processes and data collection processes.

Each letter (**A**, **B**, etc.) identifies a process.



Continued...

A sample template (continued)

Step	Process	Type	Start Trigger	End Trigger
1	A	Monotonic[1]	Step Start The first process in a step must start with the Step Start trigger.	None A "None" means that the process will complete its task.
	B	Peak/Valley Data[2]	Process A The application will start storing peak/valley data when Process A is complete.	See next page
	C	Cyclic	Process A Process C starts when Process A is complete.	See next page
	Step Done	required[3]	Specifies at least 1 and up to 8 processes that can cause the step to end. In this case, processes B and C can be selected to trigger the step done process. The completion of either process causes the step to end.	
2	D	Level Data	Step Start The application will start storing level data when Process D starts.	Process G This end trigger means that data acquisition will stop when G is completed.
	E	Monotonic	Step Start	None
	F	Cyclic	Process E	None
	G	Monotonic	Process F	None
	Step Done	required3	Specifies at least 1 and up to 8 processes that can cause the step to end. In this case, process G can be selected to trigger the step done process. Any other process would cause the remaining processes to be skipped.	

1 Test command processes (such as ramps, cycles, etc.) must be arranged serially. The end of one test command process starts the next test command process.

2 Data collection and event detection processes (such as data acquisition and limit detector) must be arranged in parallel with the test command processes. The data processes can be used to end a test command process when a specific data requirement is met.

3 The Step Done process is automatically included with every step.

Introduction

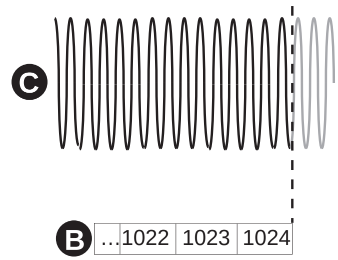
Using end triggers

The chart on the previous page did not specifically define end triggers for **B** and **C**. These two pages illustrate how you can use (or misuse) one process to stop another process.

Example 1

Example 1 assumes that process **B** (data acquisition process) is set up for 1024 peak/valley samples and process **C** (cyclic process) is set up for 2000 cycles. Process B records the peaks and valleys of the first 512 cycles of process C.

End Triggers: B=None
C=Process B



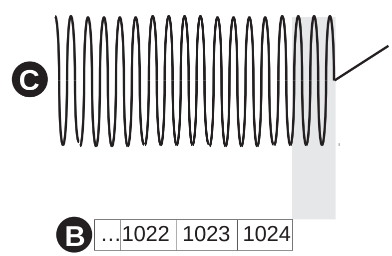
B continues to its normal termination (1024 samples).

The shaded area of **C** indicates a portion of it will not execute because the completion of **B** stopped it.

Example 2

Example 2 assumes the same setup as example 1 except that the end triggers are reversed for process **B** and process **C.**

End Triggers: B=Process C
C=None



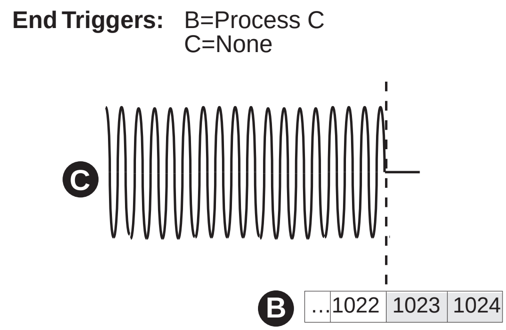
Here the data acquisition process ends before the cyclic process is complete.

C continues to its normal termination (2000 cycles).

The shaded area indicates a portion of the test that will not have data stored because **B** ran out of buffer space.

Example 3

Example 3 assumes that process **B** (data acquisition process) is set up for 1024 level crossing samples and process **C** (cyclic process) is set up for 1022 cycles. Process B records each cycle of process C.

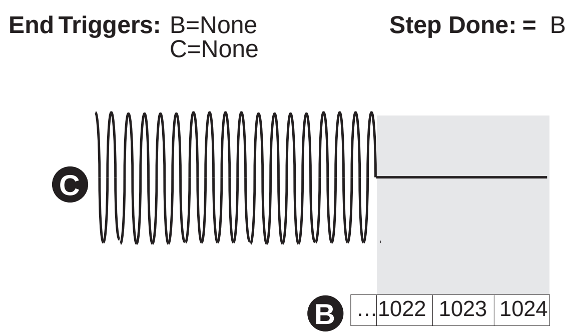


C continues to its normal termination (1022 cycles). This stops **B**, even though it hasn't reached 1024 yet.

The shaded area of **B** indicates where data will not be taken since this process is stopped.

Example 4

Example 4 assumes the same setup as example 3 except that the step done process has only the data acquisition process (B) selected. Process C should be selected to end the step.



Here the cyclic process ends before the data acquisition process is complete.

C continues to its normal termination (1022 cycles). But B hasn't reached 1024 yet.

The shaded area of B indicates where cyclic data will no longer be taken since C is stopped.

In fact, the test is trying to sample level crossing data, the test will hang (since there is no change in the level so no samples can be acquired).

Introduction

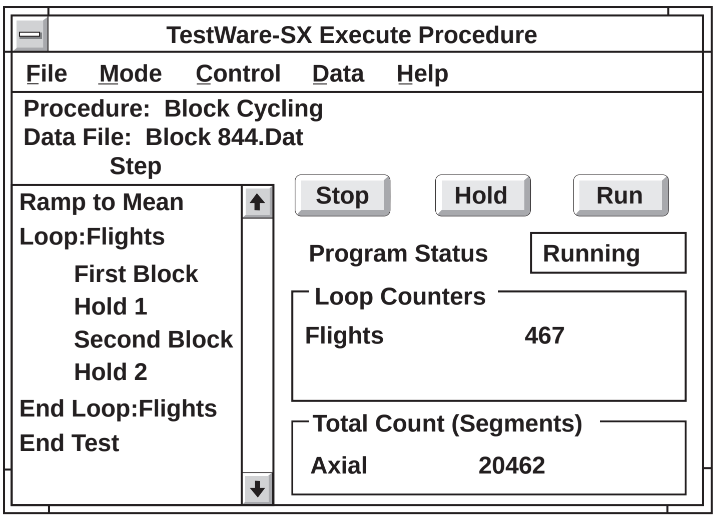
Looping Steps

The application provides the capability to loop through a series of steps. This permits more sophisticated waveform sequences, such as:

- block loading
- trapezoidal waveforms
- irregular waveforms
- repeated spectrum sequences

Note *Loops can be nested within each other.*

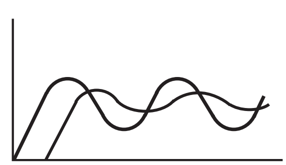
The window below shows a typical loop. Note that the loop can be assigned a name and its status displayed during a test.



Using processes in multi-axial control

The application can operate either with uniaxial or multiple axes of control. If separate control axes are running independently of each other, then each channel has its own command process.

If the axes must maintain a phased relationship with each other, then you can select a single command process and specify the desired phase angle between the channels



Creating a Test Procedure

The default test procedure

When you create a test template, it becomes the default test procedure with default values applied. This default procedure has zeros for all parameters that require a value and the first selection from every parameter list.

You have the option to assign values to the parameters of the default test procedure while creating the template. Normally you design the template and define the parameters of the default test procedure. This allows you to edit the template without losing any related test procedures.

Adding specific parameters

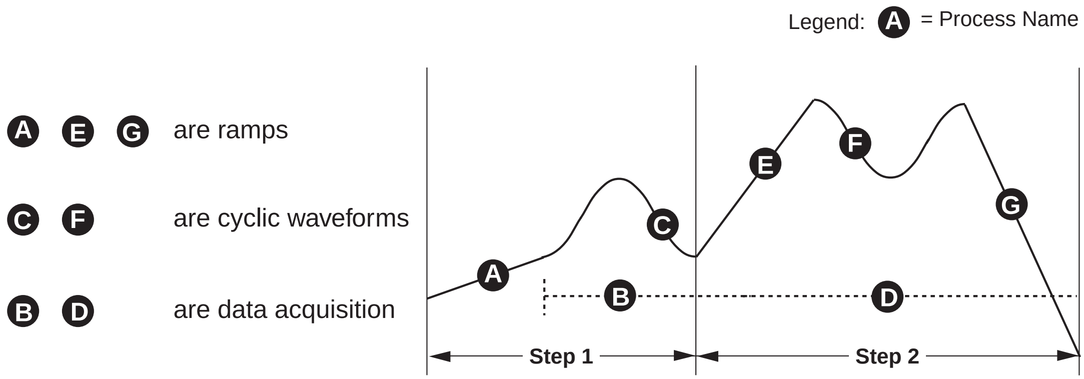
After the template has been designed, all that is needed is to assign specific values to the basic commands that have been selected. To create a specific procedure, you need to add:

- segment shape (such as ramp, haversine, or step)
- control mode (the sensor whose feedback is controlling the system)
- end level(s)
- units of measure (inches, centimeters, etc.)
- data acquisition information (such as the buffer size)
- event detection information (such as a limit value)

Continued…

Adding specific parameters (continued)

The diagram below illustrates a simple template of two steps (as described on pages 71 and 72). Each step contains command processes and data collection processes. Each letter (**A**, **B**, etc.) identifies a process.



The chart shows typical types of information added to the template (created in the previous section) to define a procedure. *Note that the template itself is not changed*—a procedure has been created *from* the template. Additional procedures can easily be created from the same template.

STEP	PROCESS	PROCESS	DESCRIPTION
1	**A**	Monotonic	Ramp to 2 cm at 1 cm/s in displacement control.
	B	Peak/Valley Data	Gather peak/valley data from the force channel.
	C	Cyclic	Two haversine segments cycle between +5 cm and +1 cm at 1 Hz in displacement control.
	Step Done	required	When the selected process is complete, the step ends.
2	**E**	Monotonic	Ramp to 4 cm at 1 cm/s in displacement control.
	D	Level Data	Record force each 0.5 cm of displacement.
	F	Cyclic	Four haversine cycles between +6 cm and +2 cm at 10 Hz in displacement control.
	G	Monotonic	Ramp to 0 at 10 kN/s in force control.
	Step Done	required	When the selected process is complete, the step ends.

Introduction

Section D: Test Recovery

TestWare-SX can stop execution of a test procedure and resume it later. This means you can recover a test after a power failure or some other catastrophic failure.

There are two basic reasons that require you to recover a test:

- The test stopped under controlled circumstances. *For example*, an interlock inadvertently stops the test because it is set to an inappropriate value or a wrong action; or, you pressed the Stop button.
- The test stopped because of a catastrophic event. *For example*, a power failure or an interlock shuts down the system.

Note *If you operate a test created with version 1.3 software, you need to enable the test recovery feature with the Recovery Options window (select Recovery Options in the Options menu of the Edit/Execute Procedure window).*

The default setting disables test recovery with version 2.0 software. Enabling test recovery when creating a template will enable the feature for all subsequent procedures. Any procedure can have test recovery enabled or disabled independently.

How it works

Test recovery uses test status information that is associated with a data file. The test status information is saved as a separate file using the same path and filename as the data file; the test status file includes the extension .SXS.

When the auto save feature is enabled, a “snap shot” of the test status is taken whenever the program status changes. The “snap shot” is automatic and can be configured to save the test status periodically.

Test recovery always restarts a test from the stopped state (even if the test was running during the last snap shot) unless the test was done or reset.

What is saved

Test status includes information about a test that is currently in progress. Components of the test status information are:

- A copy of the test procedure. It is possible to edit a procedure, then execute it without saving it. In this situation, the test status contains the information used in the execution, not the original procedure.
- The active step and processes. It also identifies how much of each process assignment is complete.
- The current program state (run, hold, stop). When recovering a test, if the program was running or holding, the test is restored in the stop state.
- The current loop count and the total segment count.
- The Test Description entered with the Description selection in the Data menu.
- The current data file.

What is not saved

When you recover a test, the test status is restored to the point where the last snap shot was taken.

- If the test is stopped by an uncontrolled event, any data acquired from the last snap shot to the point of the shutdown is lost.
- If data was being transferred from the data buffer to the computer, data that had not been transferred is lost.

Recovery accuracy

There are two measures of "accuracy" associated with a saved test status:

How old is the snap shot?

> *For example*, if the snap shot is taken one minute before the test stops, many additional cycles may be applied to the specimen that the test status would not know about.

How consistent is the snap shot?

> *For example*, if the snap shot is taken while the test is running, each active process is checked in a sequence that results in test status data that is taken at slightly different times.

Review the following characteristics of the test recovery feature to understand the advantages and disadvantages of recovering a test under different circumstances.

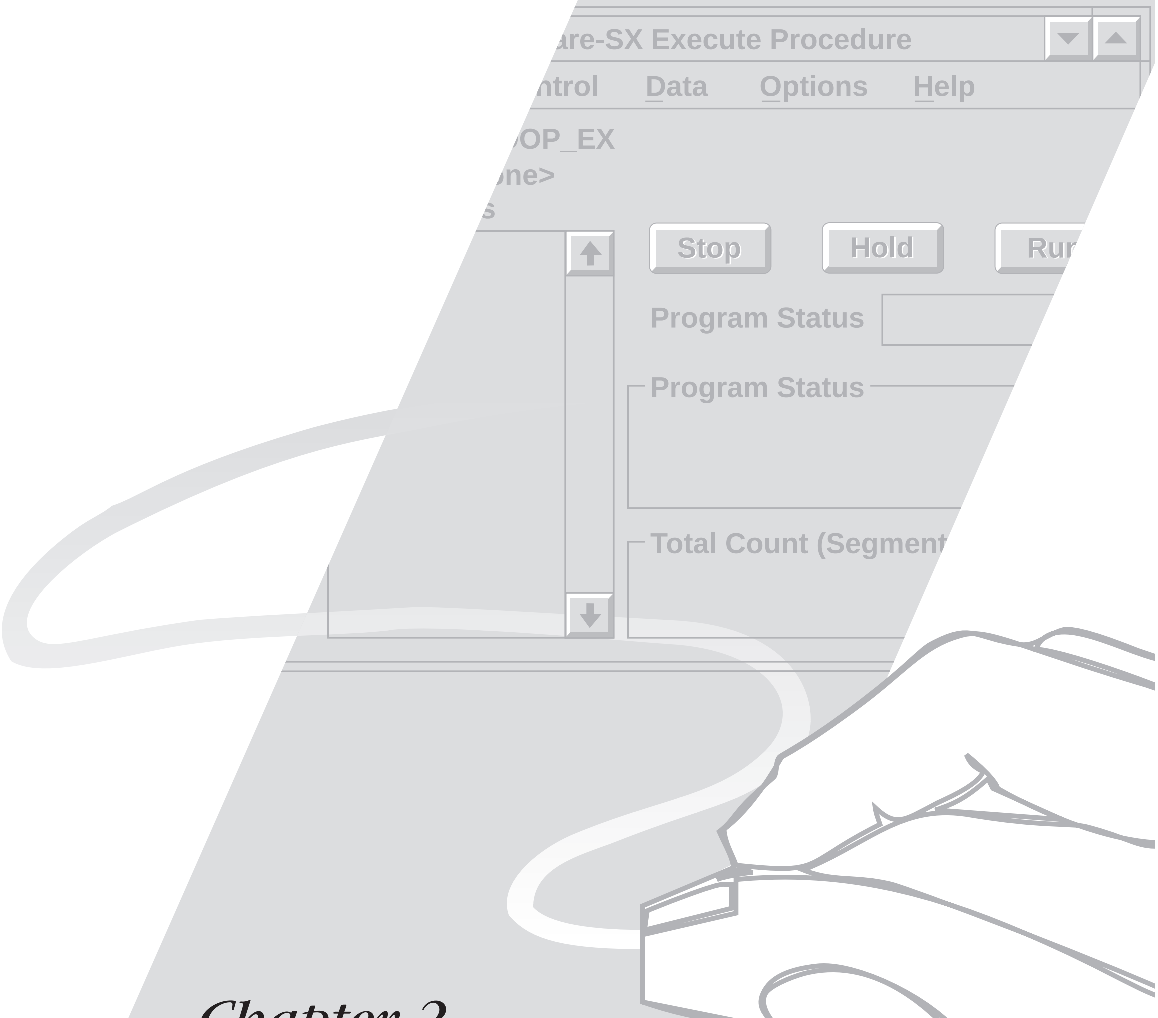
- The auto save feature saves the test status any time the program status changes from run to stop or stop to run (including interlocks). The auto save feature has two options: saving at the end of a step and saving at time intervals.
- Recovering a test that was saved at the end of each step provides very consistent test status data because all processes are inactive at the end of a step. The test begins with the next step. However, if the test was stopped in the middle of a step, the information from that half step is lost.
- Recovering a test that was saved periodically provides test status data that is relatively close to the point the test stopped. The shorter the period, the closer the test status is to the point of failure. However, saving a test does require additional processor time. If the auto save rate is too fast, the accuracy of the data acquisition processes is adversely affected (data output and response of triggers).

Note *One test status of the procedure is maintained. Each auto save action overwrites the test status. If the system crashes (power fails) while saving of test status is in progress, you probably cannot recover the test status!*

MTS®

TestStar™ II

Control System



Chapter 2
Designing a Test

Create a Test Template

- To establish a new test template
- To define the default test parameters

Open the Edit Template window

Add the required steps

Add the required processes

Define each process (optional)

Define the step done process

Save the test template

Guidelines

Draw the waveform to help identify the required steps and processes.

Be sure at least one process uses the step start trigger for each step.

Be sure at least one process is selected for the step done process.

Review the start and stop triggers for all prcesses for proper sequencing.

Arrange the list of processes in the order of execution.

Data processes should be listed before the corresponding command process.

Define a Test Procedure

- To correct a new test procedure
- To define a new procedure with different parameters for a template

Open a template

Open the Edit Procedure window

Select a step

Select a process, change any parameter

Save the procedure as a different name

Edit a Test Template

- To correct a new test template
- To create a new template similar to an existing template

Open a template

Add or edit a step

Add or edit a process

Save the template as a different name

Chapter 2

Designing a Test

Contents

Overview

This chapter describes how to design and create a test in the TestWare-SX application.

Section A illustrates how to make a test template and define the default test procedure for the template.

Section B illustrates how to make additional test procedures from a template.

Section C illustrates how to change an existing template and save it as a different template.

The following describes the primary differences between a template and a procedure.

	Template	Procedure
Purpose	Creates a sequence of processes by adding steps and processes.	Defines the parameters for each process.
Process Information	Defines when the process starts and when it stops. Defines what control channels the process is to be applied. Names the process.	For command processes, selects the control mode (for each control channel), the types of units (to set the parameters), command amplitude, and segment shape. For data processes, selects type of data to be monitored, how the data is monitored, how data is stored, etc.
File Name	The template is the default procedure with the file extension .000.	Each procedure has a unique file extension within the range of .001 to .999.

Section A: Creating a Test Template

This section describes how to make a test template. To illustrate the technique to create a test template we designed an example test.

In this Section

Guidelines

Be sure at least one process uses the step start trigger.

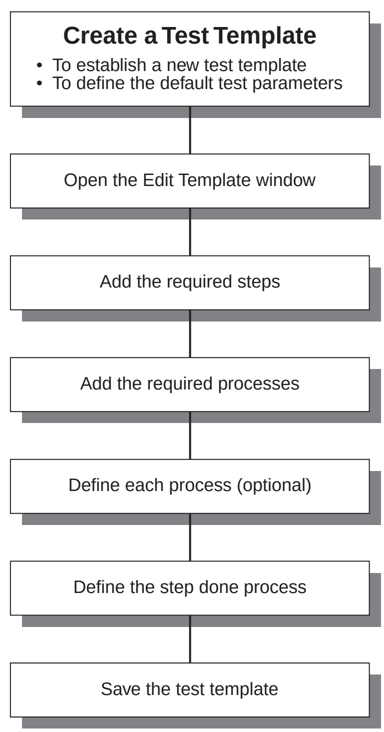
Be sure at least one process is selected in the Step Done window.

Review the start and end triggers for all the processes. Sometimes when a process is added, the desired end trigger is added later.

The list of processes should also reflect the order they are started.

There can be a slight delay between multiple processes when they use the same start trigger. The process listed first begins first.

Observe the following guidelines when you create a test template:



Task 1 Make a Drawing of Your Test Program

Before you can use the TestWare-SX application you need to know what your test requirements are.

Procedure

Step 1 Determine the test requirements

The example in this section has two primary goals:

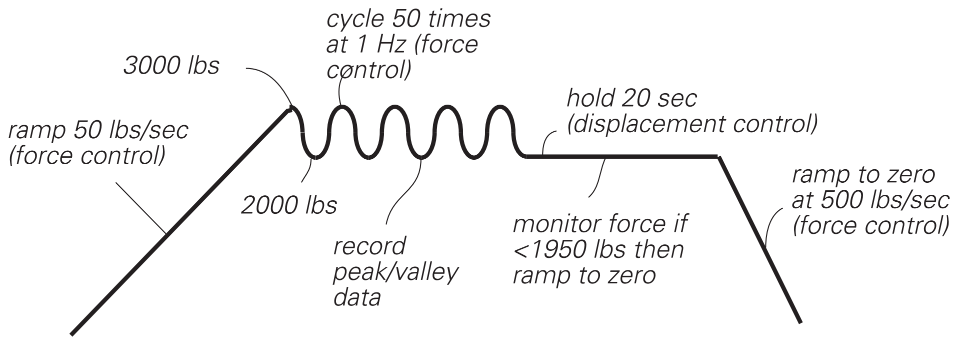
- Cycle the specimen between 2000 and 3000 pounds while collecting peak/valley data.
- Hold the specimen in length control while waiting for a 50 pound drop from the force sensor.

After the primary information is determined, additional information may be needed to complete a test definition. The example has two additional components:

- A ramp from a starting position to the cycling level.
- A ramp from the finishing position back to the starting position.

Step 2 Draw the waveform

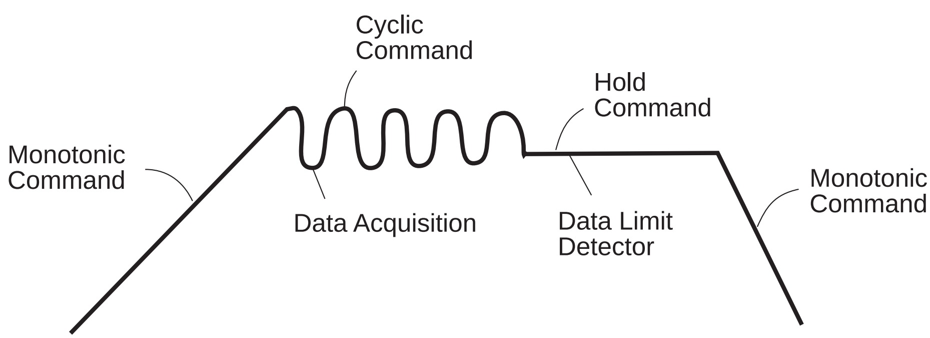
Sketch the waveform design and identify significant information.



The waveform drawing is not to scale. It only shows specific information that will be needed to create the test program.

Step 3 Identify the required processes

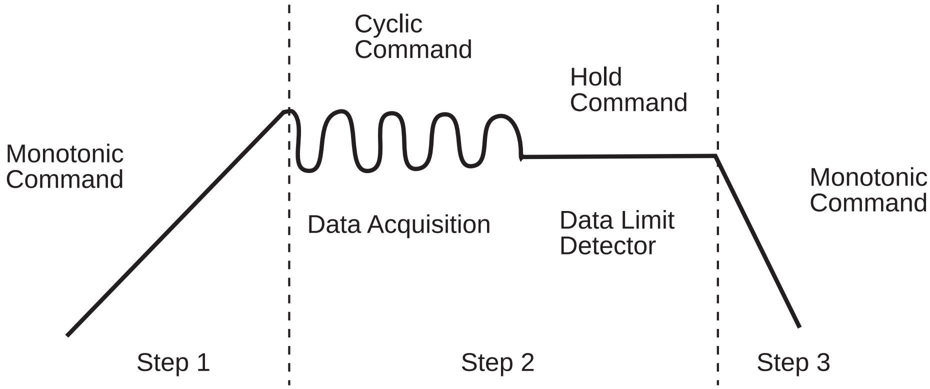
In order to identify specific processes you need to be familiar with the available processes.



The required processes are shown for each test requirement.

Step 4 Identify groups of processes for steps

The example test program could be entered as one step. A single step for this short test program would be appropriate. However, to illustrate how to use steps, the example uses three steps to construct the waveform.



Section C, Editing a Template, describes how to repeat Step.

Task 2 Open the TestWare-SX Application

Procedure

Step 1 Start the TestWare-SX application program

From the OS/2 window, open the TestStar folder, then double-click the TestWare-SX application program.

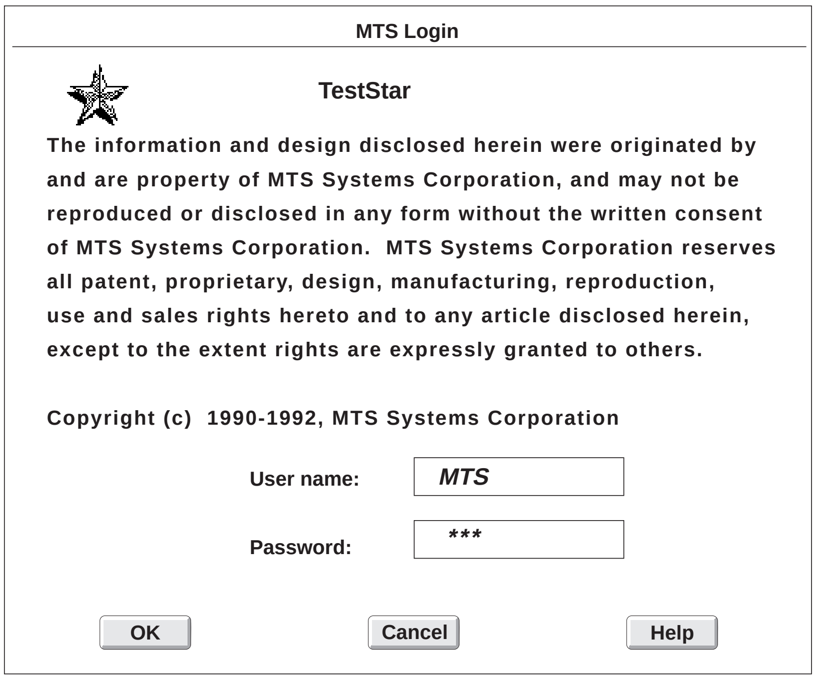
Step 2 Log on as a user

Type in your user name and password. Press OK when you have completed these two fields.

- The password is not displayed; instead, asterisks are used for password security.
- The program initializes itself based on the User name you enter.
- When you press OK, the main TestStar window appears.

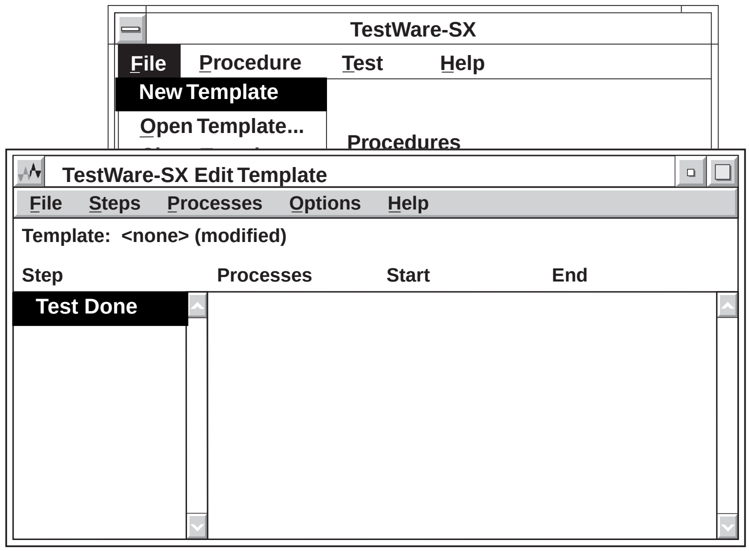
When you log on, the system automatically opens your configuration file.

See Chapter 8 in the TestStar Reference manual to change your default configuration.



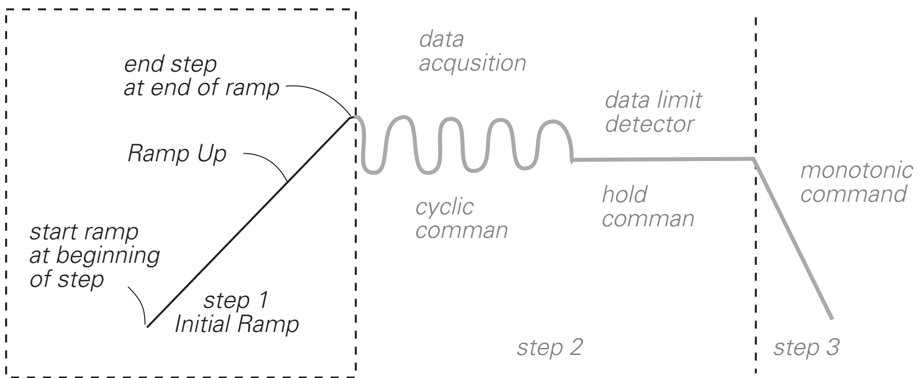
Step 3 Open the Edit Template window

Select New Template in the File menu of the TestWare-SX window to open the Edit Template window.



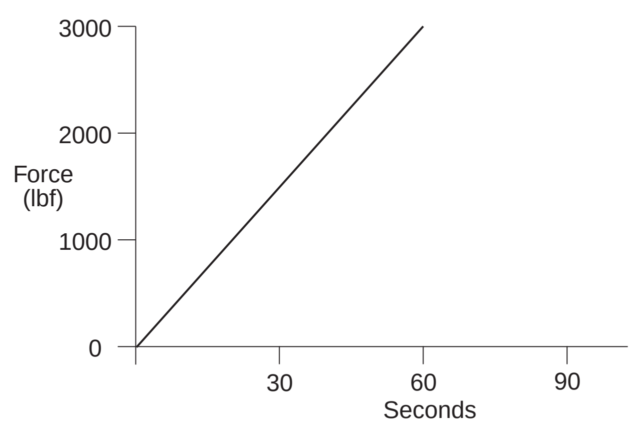
Task 3 Create Step 1

Procedure



Creating Step adds the first step of the test template. The step consists of a single process.

Step 1 produces a 60-second ramp to 3000 lbf.

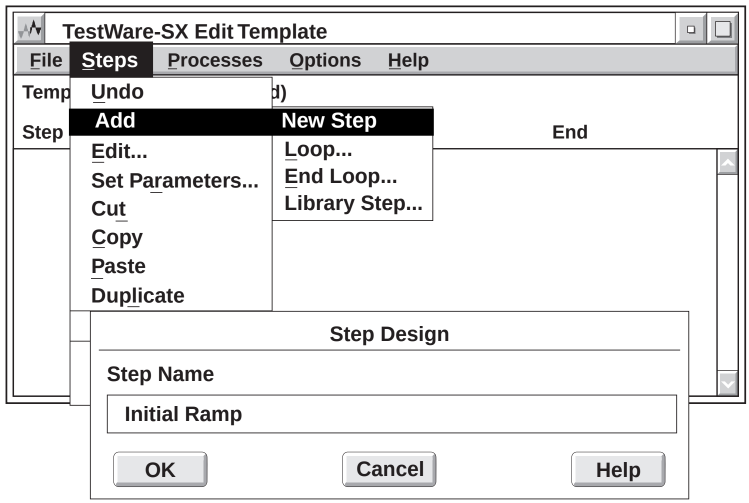


Step 1 Add Step 1

A Select the Steps menu in the Edit Template window.

B Select Add in the Steps menu.

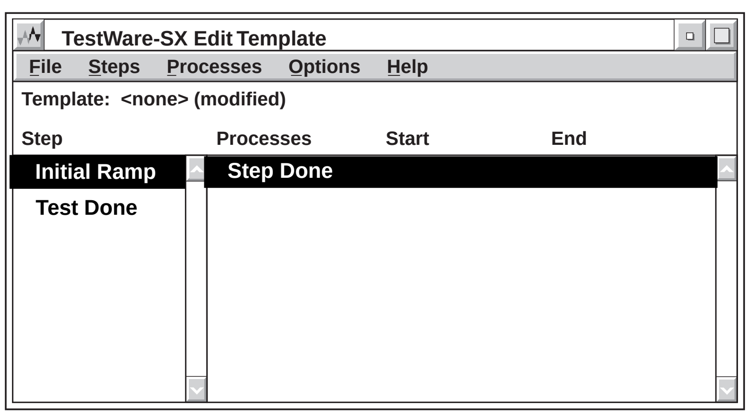
C Select New Step in the Add sub menu.



D Enter a name you wish to call the step. In this example the step is called **Initial Ramp**.

After you enter the step name, it is listed in the steps column.

Also, Step Done is added to the Processes list and the Processes menu is enabled.



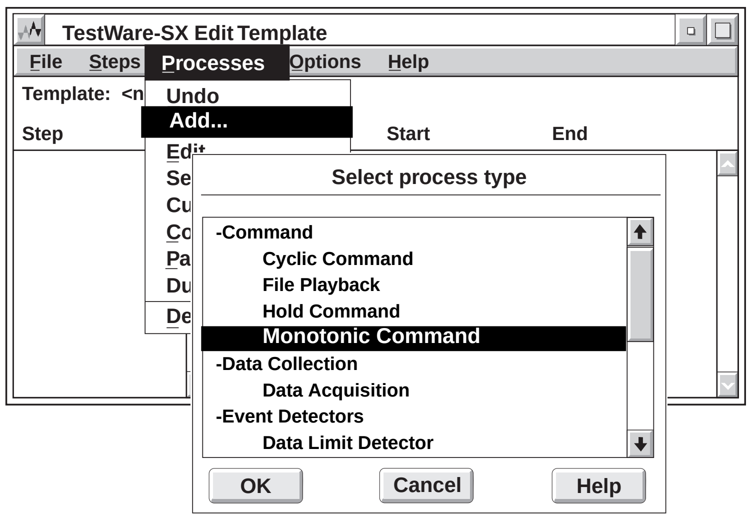
Designing a Test

Step 2 Add a monotonic command

A Select the Processes menu in the Edit Template window.

B Select Add in the Processes menu.

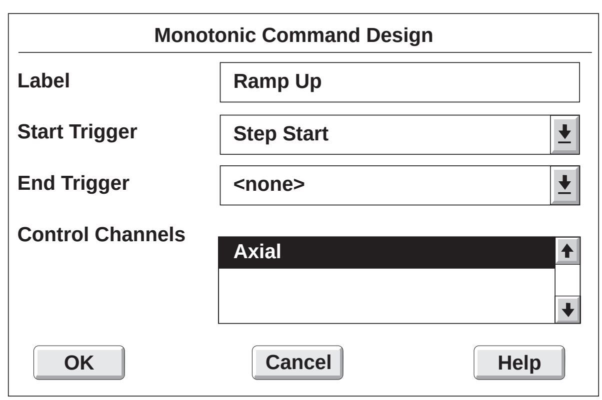
C Select Monotonic Command in the Select process type window and press the OK pushbutton.



D Complete the Monotonic Command Design window as shown and press the OK pushbutton.

The Ramp Up process starts at the beginning of the step (Step Start).

The Ramp Up process ends at the end of the step (<none>).



Step 3 Define the monotonic command

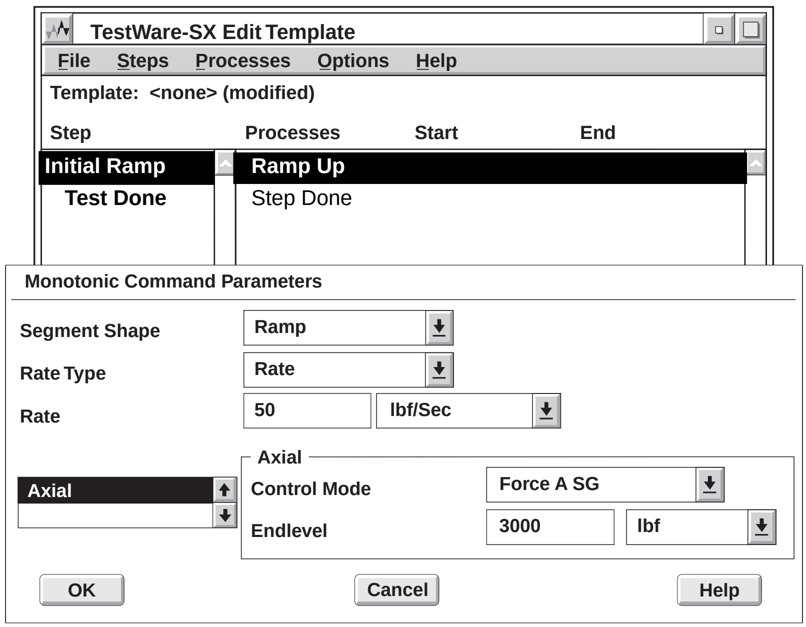
A Double-click Ramp Up in the process list to display the Monotonic Command Parameters window.

Note *The Monotonic Command Parameters window can also be opened by selecting the Ramp Up process and selecting Set Parameters in the Processes menu.*

The Ramp Up process produces a ramp from zero (the starting position) to 3000 lbf at a rate of 50 lbf/second.

Defining some parameters affects the selections of other parameters.

Be sure to define the parameters in the order given below.



Complete the window in the following order:

B Select the Axial control channel.

C Select the Force control mode.

D Select lbf as the end level units.

E Select Rate for the rate type.

F Enter the remaining parameters.

Step 4 End Step 1

A Select (highlight) the Step Done process.

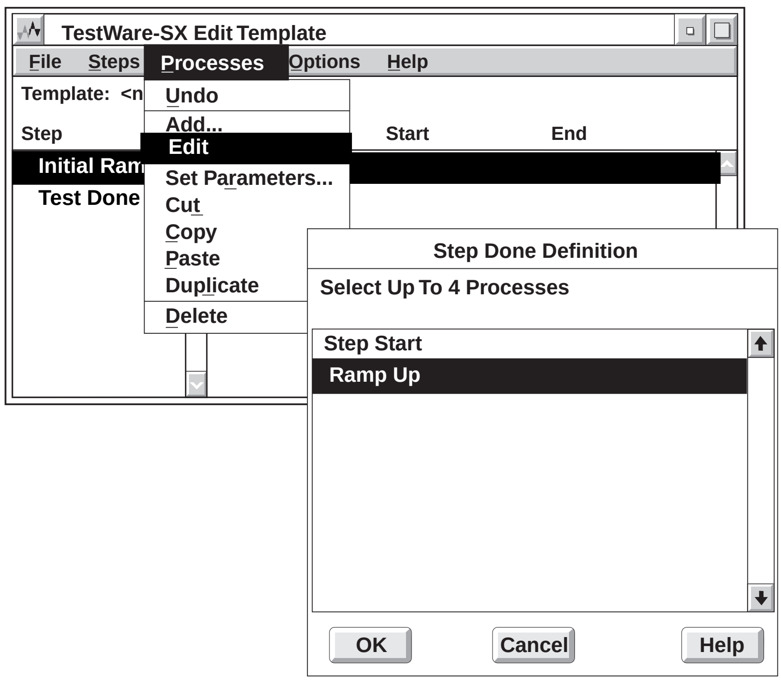
B Select Edit in the Processes menu or double-click on the right mouse button.

C Select (highlight) the Ramp Up process and press the OK pushbutton.

Note *Selecting Step Start in the Step Done Definition causes the step to end when it begins. Always select a process within the step.*

Be sure you have Step Done highlighted when you select Edit in the Processes menu and double--click on the right mouse button.

At least one process must be selected for the step done definition.

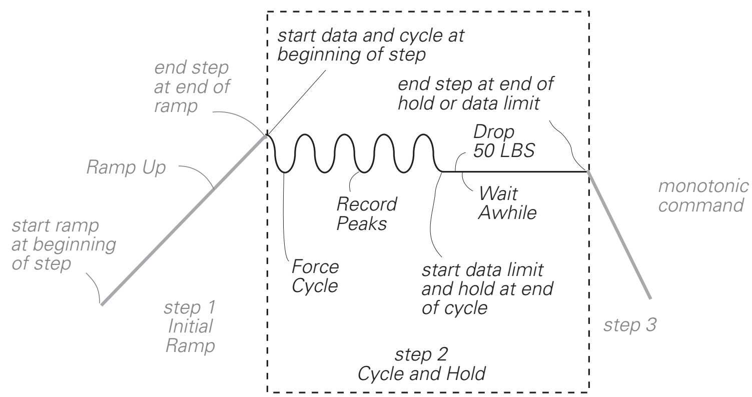


Designing a Test

Task 4 Create Step 2

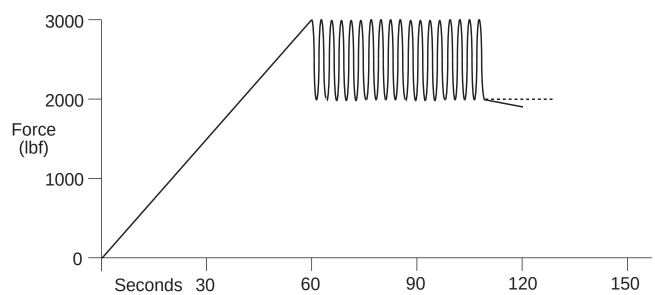
Procedure

Creating Step adds a second step with four processes to the template.



Step produces 50 cycles while acquiring peak/valley data, then holds the actuator waiting for a drop of 50 lbf.

Either a drop of 50 lbf or the 20-second time-out will end the step.



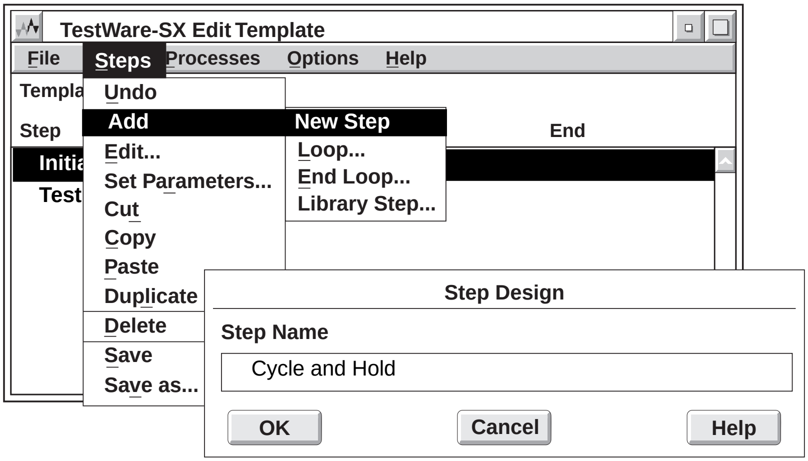
Step 1 Add Step 2

A Select (highlight) Test Done in the Steps list. Step will be inserted above Tet Done and below Initial Ramp.

B Select the Steps menu in the Edit Template window

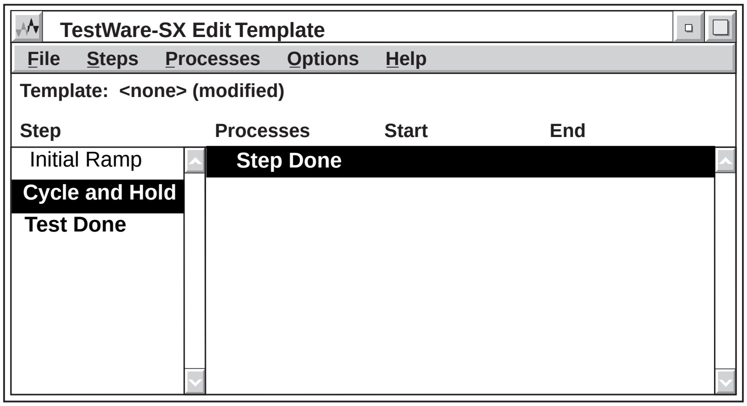
C Select Add in the Steps menu.

D Select New Step in the Add sub menu.



Enter a name you wish to call the step. In this example the step is called **Cycle and Hold**.

The Cycle and Hold step is inserted above Test Done and below Initial Ramp step.

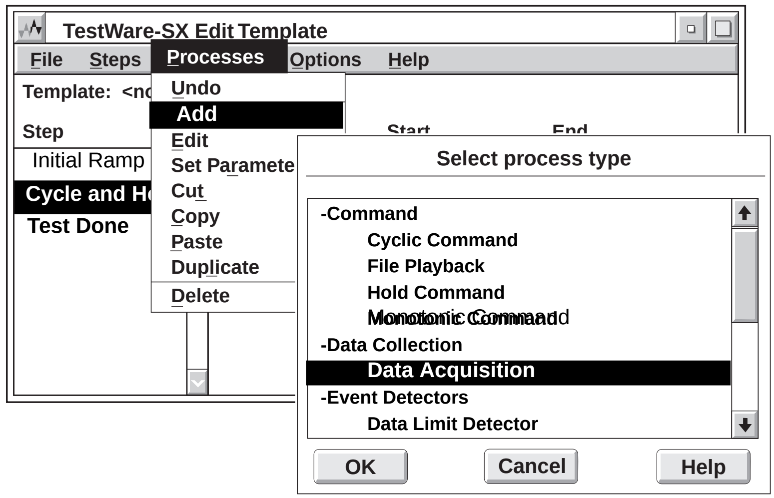


Step 2 Add a data acquisition process

A Highlight the Step Done process and select the Processes menu in the Edit Template window.

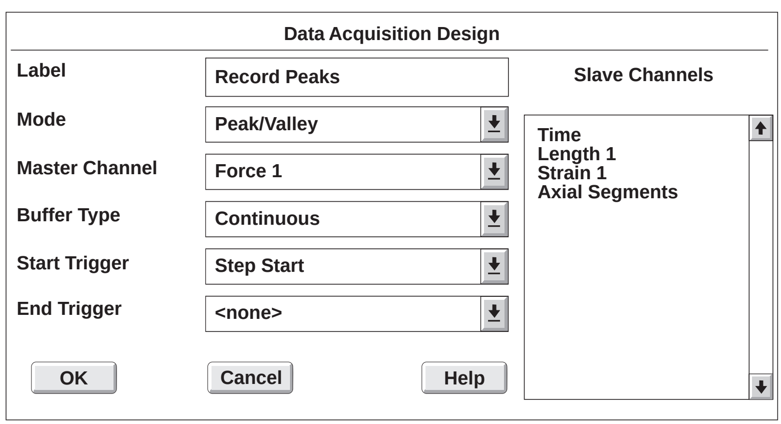
B Select Add in the Processes menu.

C Select Data Acquisition in the Select process type window and press the OK pushbutton.



D Complete the Data Acquisition Design window as shown and press the OK pushbutton (none of the slave channels are selected).

This process starts at the beginning of the step (Step Start). The stop trigger (<none>) will be changed after the cyclic process is added (which will become the stop trigger).

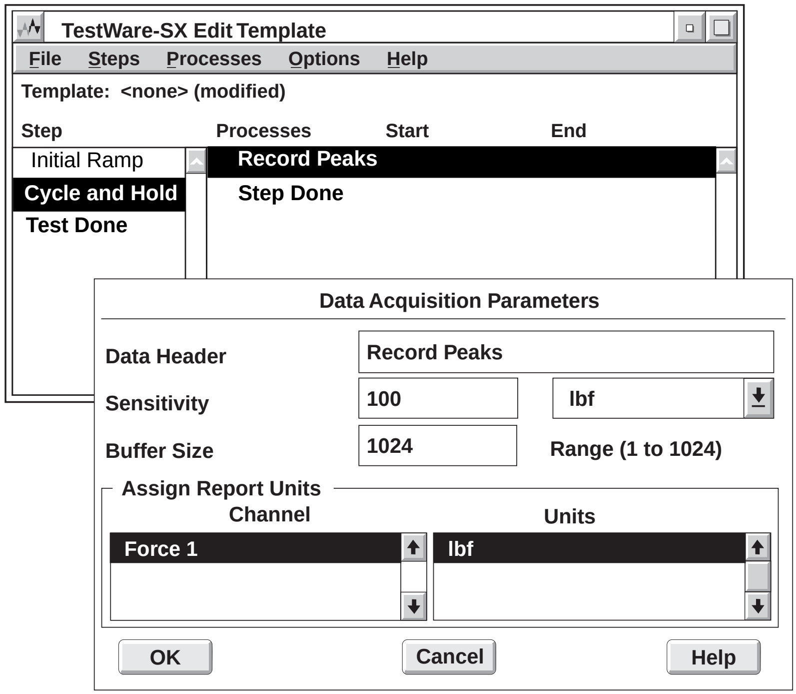


Step 3 Define the data acquisition process

A Double-click Record Peaks in the process list to display the Data Acquisition Parameters window.

Note *The Data Acquisition Parameterndow can also be opened by selecting the Record Peaks process and selecting Set Parameters in the Processes menu.*

The Record Peaks process monitors the force channel and records any peaks and valleys that change at least 100 lbf.

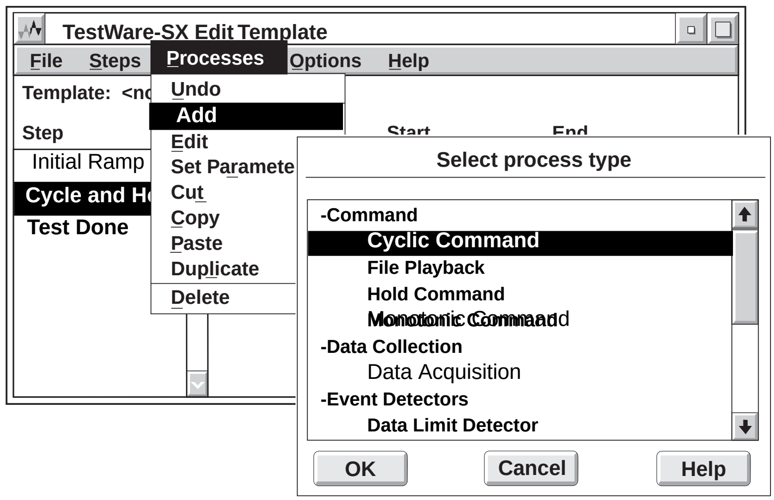


Step 4 Add a cyclic command

A Highlight the Step Done process and select the Processes menu in the Edit Template window.

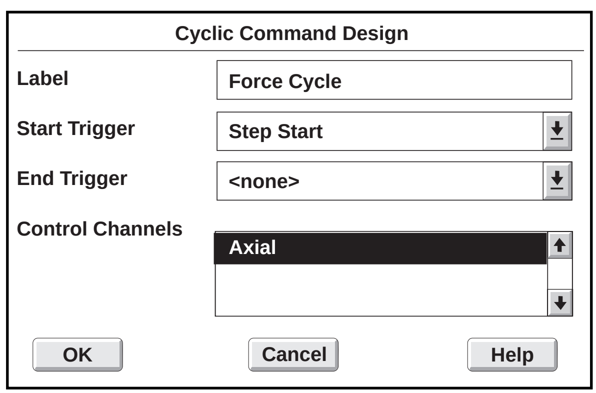
B Select Add in the Processes menu.

C Select Cyclic Command in the Select process type window and press the OK pushbutton.



D Complete the Cyclic Command Design window as shown and press the OK pushbutton.

This process starts at the beginning of the step (Step Start). The command ends at the end of the step (<none>).

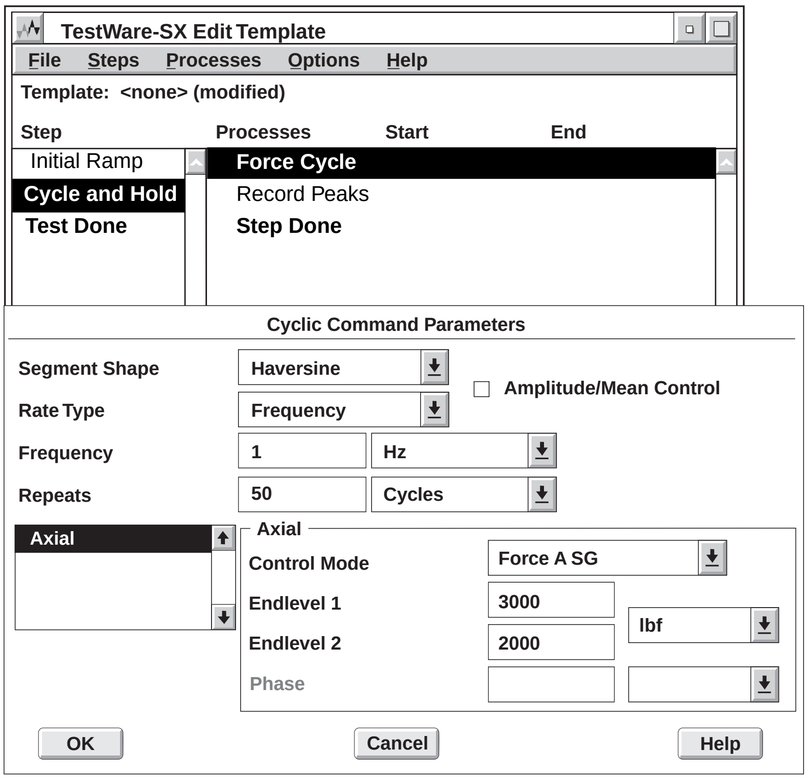


Step 5 Define the cyclic command

A Double-click Force Cycle in the process list to display the Cyclic Command Parameters window.

Note *The Cyclic Command Parameters window can also be opened by selecting the Force Cycle process and selecting Set Parameters in the Processes menu.*

The Force Cycle process cycles between 2000 and 3000 lbf for 50 cycles.



Perform the following in order.

B Select the Axial control channel.

C Select force as the control mode.

D Select lbf as the units for the end levels.

E Enter the remaining parameters as shown.

Step 6 Edit the data acquisition process

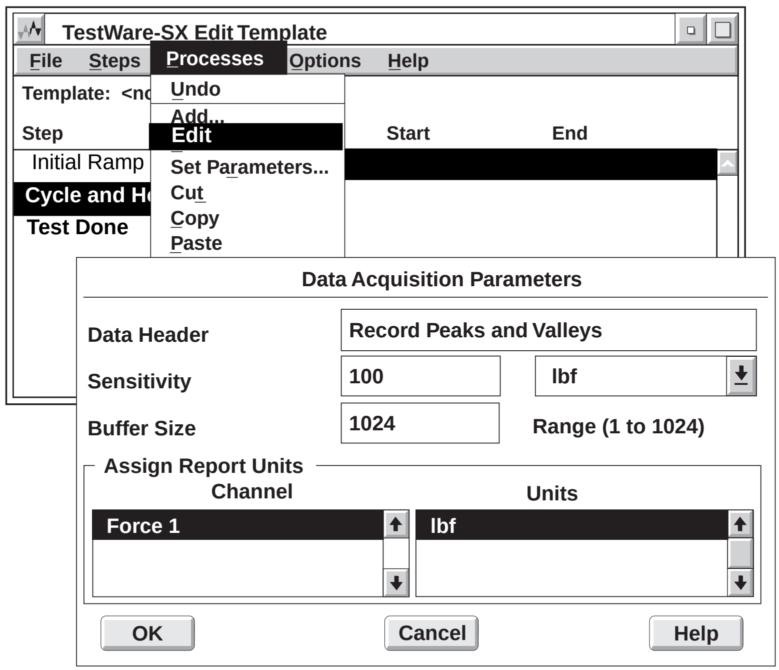
Edit the data acquisition process to select the cyclic command process as the stop trigger.

A Highlight the Record Peaks process.

B Select Edit in the Processes menu.

C Change the Stop Trigger from <none> to Force Cycle and press the OK pushbutton (refer to Step).

The Record Peaks process ends when the Force Cycle process ends.

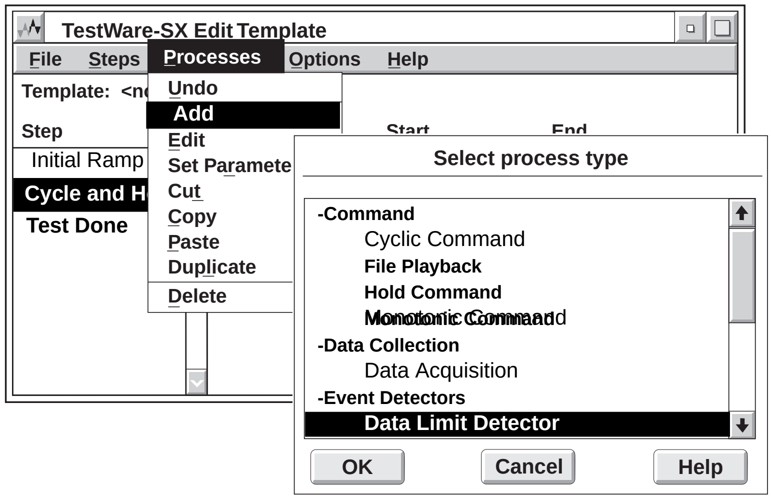


Step 7 Add a data limit detector

A Highlight the Step Done process (in the Process list) and select the Processes menu in the Edit Template window.

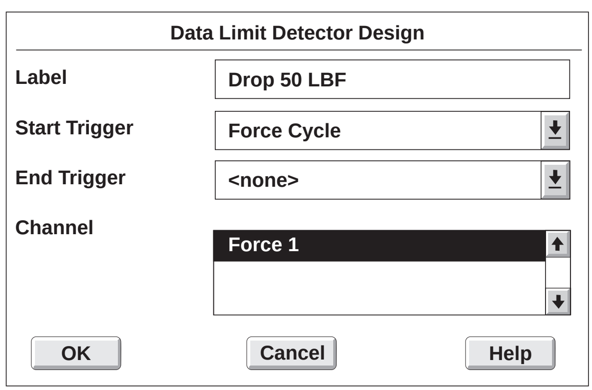
B Select Add in the Processes menu.

C Select Data Limit Detector in the Select process type window and press the OK pushbutton.



D Complete the Data Limit Detector Design window as shown and press the OK pushbutton.

This process starts when the Force Cycle process is complete. The process ends at the end of the step (<none>).

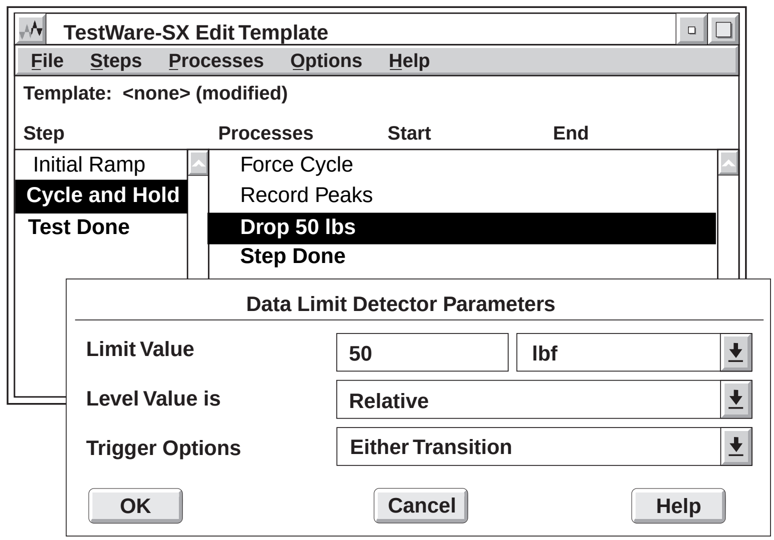


Step 8 Define the data limit detector

A Double-click Drop 50 LBF in the process list to display the Data Limit Detector Parameters window.

Note *The Data Limit Detector Parameters window can also be opened by selecting the Drop 50 LBF process and selecting Set Parameters in the Processes menu.*

The Drop 50 LBF process monitors the force channel for a change of 50 pounds.

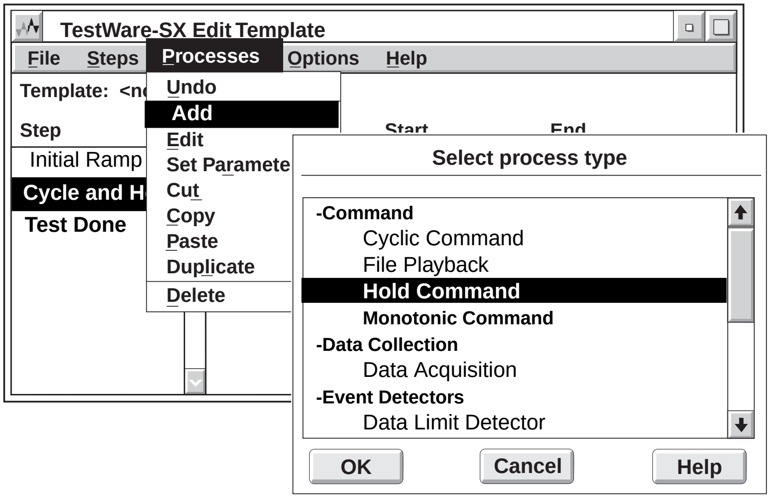


Step 9 Add a hold command

A Highlight the Step Done process and select the Processes menu in the Edit Template window.

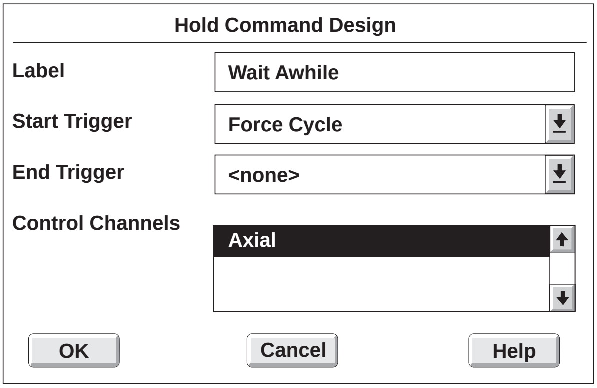
B Select Add in the Processes menu.

C Select Hold Command in the Select process type window and press the OK pushbutton.



D Complete the Hold Command Design window as shown and press the OK pushbutton.

This process starts when the Force Cycle process is complete. The process ends at the end of the step (<none>).

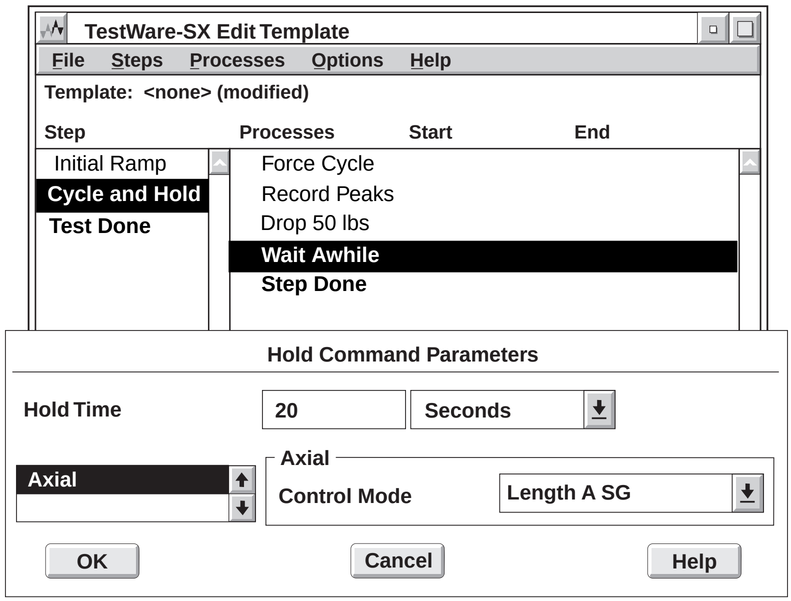


Step 10 Define the hold command

A Double-click Wait Awhile in the process list to display the Hold Command Parameters window.

Note *The Hold Command Parameters window can also be opened by selecting the Wait Awhile process and selecting Set Parameters in the Processes menu.*

The Wait Awhile process holds the actuator at its existing position for 20 seconds.



B Select the Axial control channel.

C Select length as the control mode.

D Enter 20 and select seconds in the Hold Time entry fields.

Step 11 End Step 2

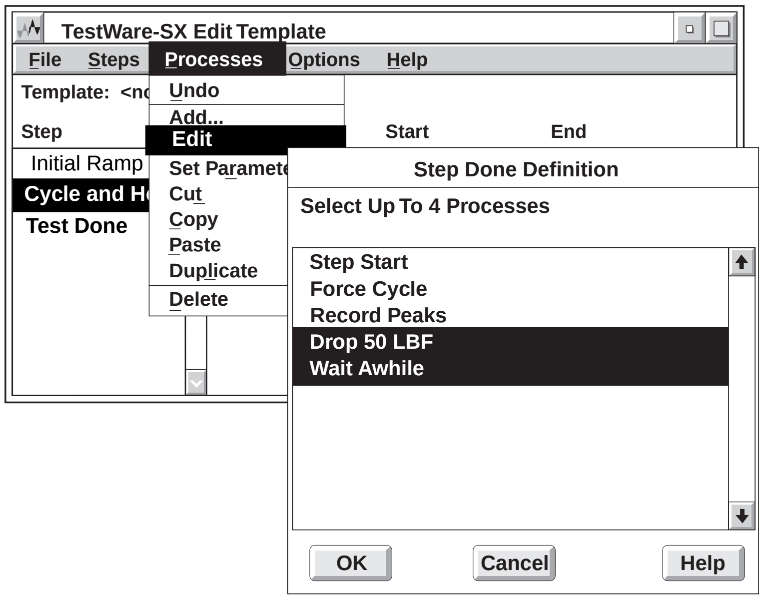
A Select (highlight) the Step Done process.

B Select Edit in the Processes menu or double-click on the right mouse.

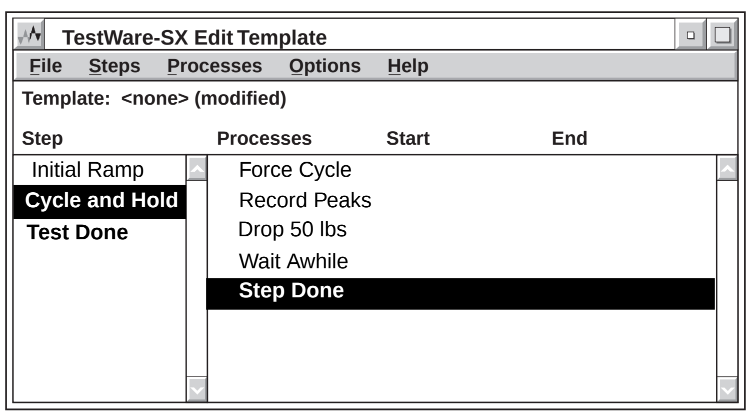
C Select (highlight) the Wait Awhile and Drop 50 LBF processes and press the OK pushbutton.

At least one process must be selected to end the step. When the selected step ends, all processes end.

Select both the Drop 50 LBF and Wait Awhile processes. If either one of these steps completes its task, the step will end.



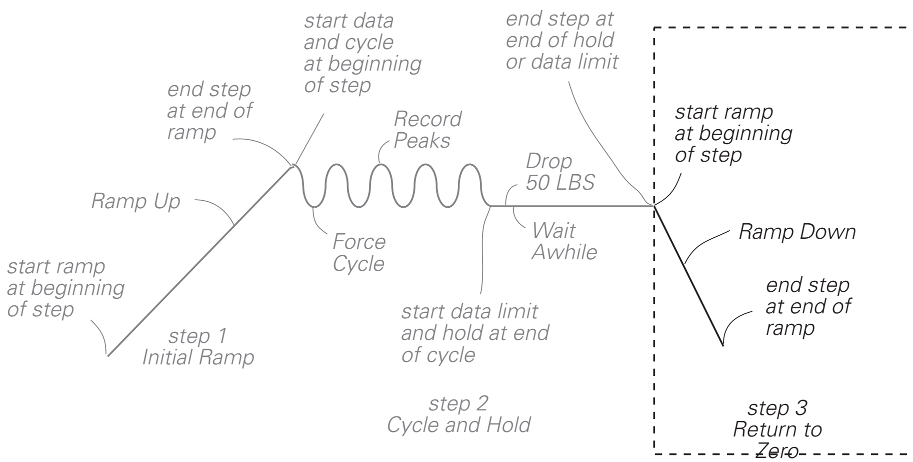
All of the processes you added are shown in the Process list.



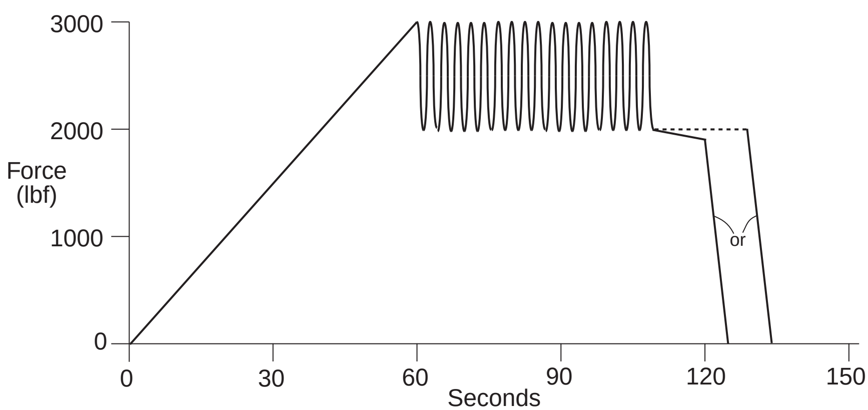
Task 5 Create Step 3

Procedure

Creating Step adds a third step with a single process to end the template.



Step ramps to zero at a rate of 500 lbf per second when Step ends.



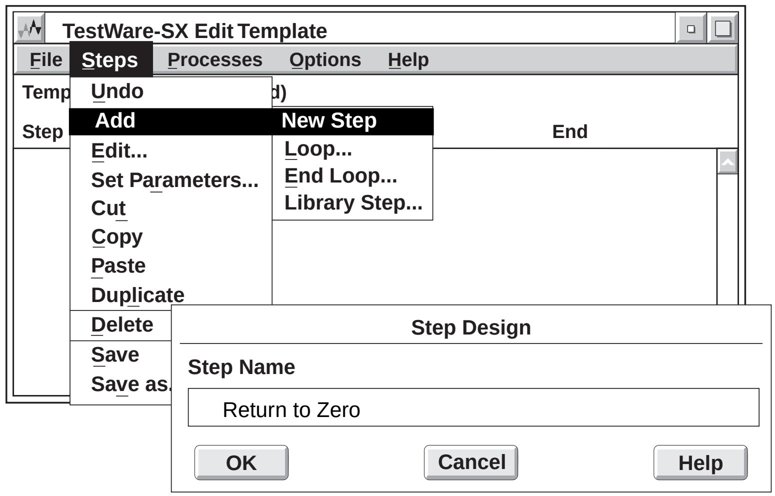
Step 1 Add Step 3

A Select (highlight) Test Done in the Steps list. Step will be inserted above Test Done.

B Select the Steps menu in the Edit Template window.

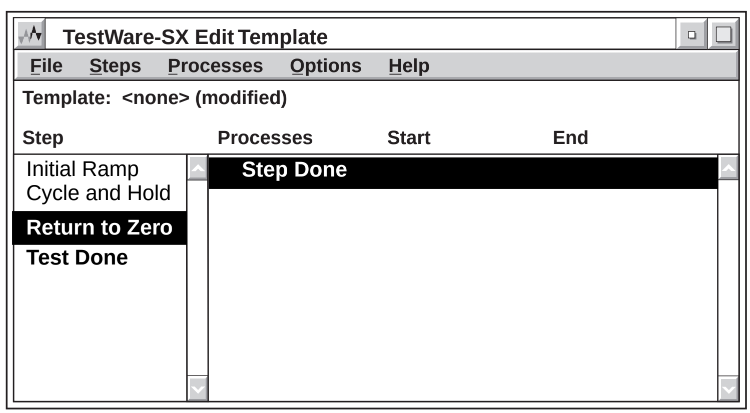
C Select Add in the Steps menu.

D Select New Step in the Add sub menu.



Enter a name you wish to call the step. In this example the step is called **Return to Zero**.

The Return to Zero step is inserted above Test Done and below Cycle and Hold step.

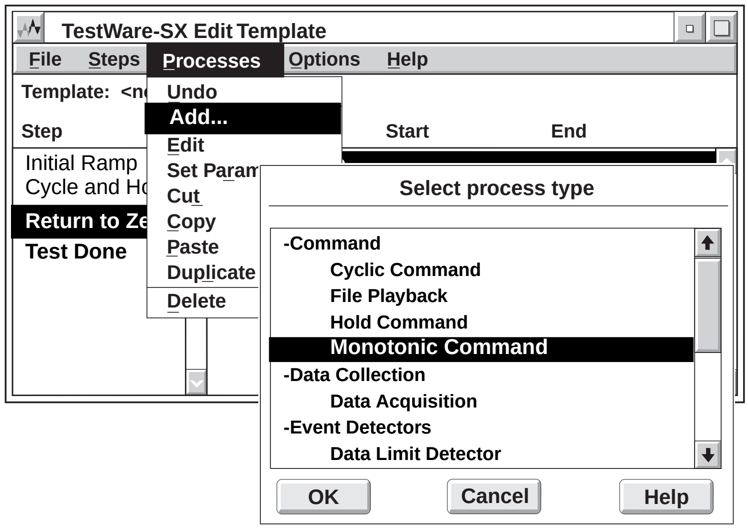


Step 2 Add a monotonic command

A Select the Processes menu in the Edit Template window.

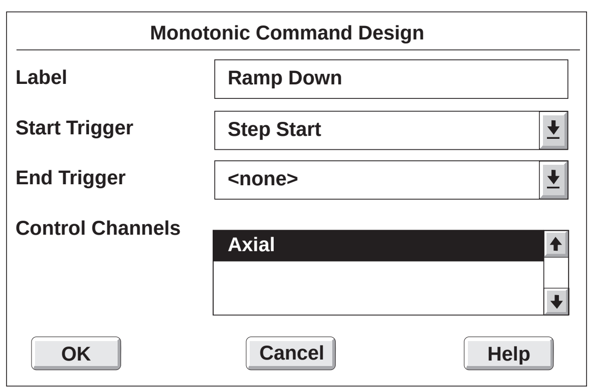
B Select Add in the Processes menu.

C Select Monotonic Command in the Select process type window and press the OK pushbutton.



D Complete the Monotonic Command Design window as shown and press the OK pushbutton.

This process starts at the beginning of the step (Step Start). The process ends at the end of the step (<none>).

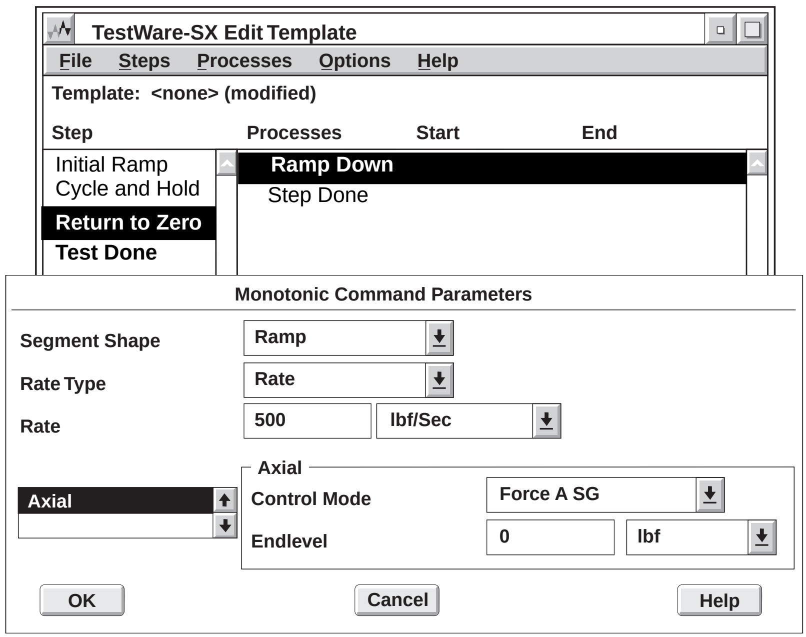


Step 3 Define the monotonic command

A Double-click Ramp Down in the process list to display the Monotonic Command Parameters window.

Note *The Monotonic Command Parameters window can also be opened by selecting the Ramp Down process and selecting Set Parameters in the Processes menu.*

The Ramp Down process produces a ramp from the force level at the beginning of the process to zero at a rate of 500 lbf/second.



Complete the window in the following order:

B Select the Axial control channel.

C Select the Force control mode.

D Select lbf as the end level units.

E Select Rate for the rate type.

F Enter the remaining parameters.

Step 4 End Step 3

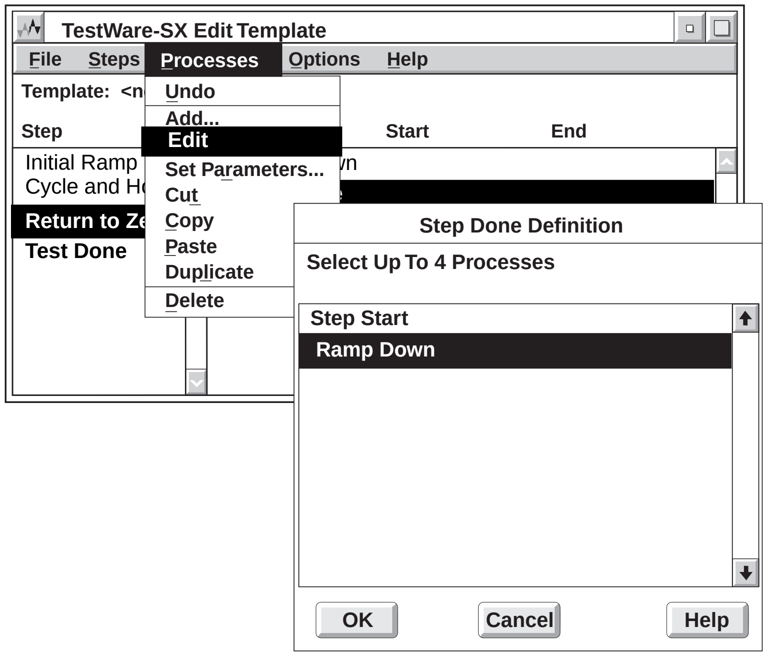
A Select (highlight) the Step Done process.

B Select Edit in the Processes menu or double-click on the right mouse button.

C Select (highlight) the Ramp Down process and press the OK pushbutton.

At least one process must be selected to end the step.

Select the Ramp Down process. When the Ramp Down process ends, Step ends.



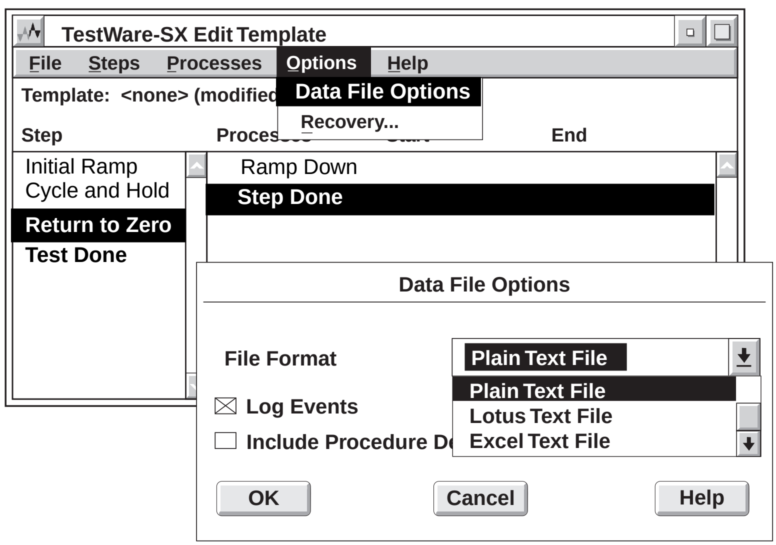
Step 5 Select the data file format

The data acquired during the test can be formatted for your preferences.

From the Edit Template window, select the Options menu, then select Data File Options.

- Enable Log Events if you want to log when the Run, Hold, and Stop pushbuttons are pressed.
- Enable Include Procedure Description if you want a listing of the test procedure steps, processes, and parameters included with the test data. The procedure description is the same information that can be acquired with the Print Preview function in the File menu.
- Select the File Format best suited for the application you intend to use with the test data.

You can select a file format for your data that is compatible with most software applications.



Task 6 Save the Template

Saving the template records your test design to disk.

A Select the File menu in the Edit Template window.

B Select Save in the File menu.

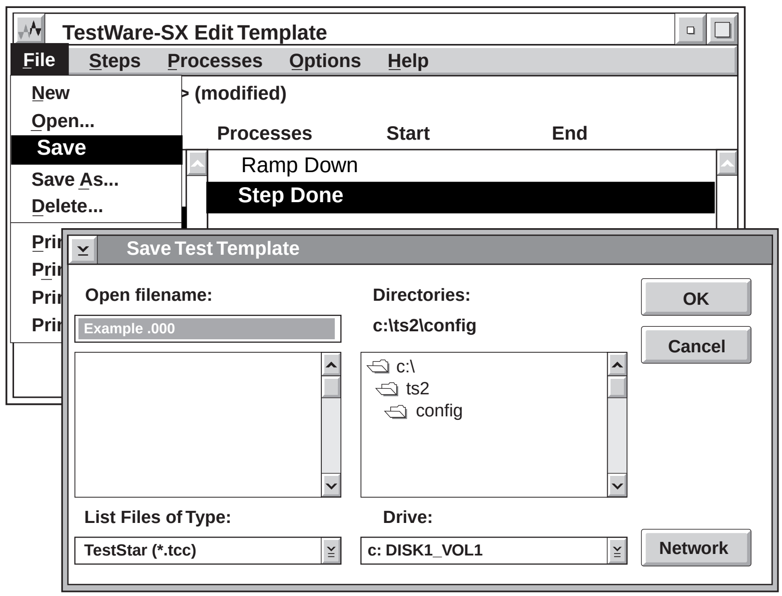
C Type the name you wish to call the template in the File entry field and press the Save pushbutton.

The default directory is located on the C: drive in the directory called TWSX, which is located in the TS directory.

The example template constructed in this section is called Example.

When the template is saved, the template name is shown in the Edit Template window.

The extension .000 is automatically added.



Saving your work periodically

If you create a long template, you may want to save your work before you complete the entire template. Selecting Save in the File menu the first time displays the Save Test Template window. After the file is saved, selecting Save again updates the file without displaying the Save Test Template window.

Section B: Editing a Test Procedure

Procedure

Prerequisite

This section uses the test template created in Section A. This section also assumes you have opened the TestWare-SX application.

Edit Guidelines

There are two reasons you would use the edit procedure function:

- To correct a procedure you already defined.
- To create a new procedure from a template.

You can change any process parameter in a test procedure with the edit procedure function.

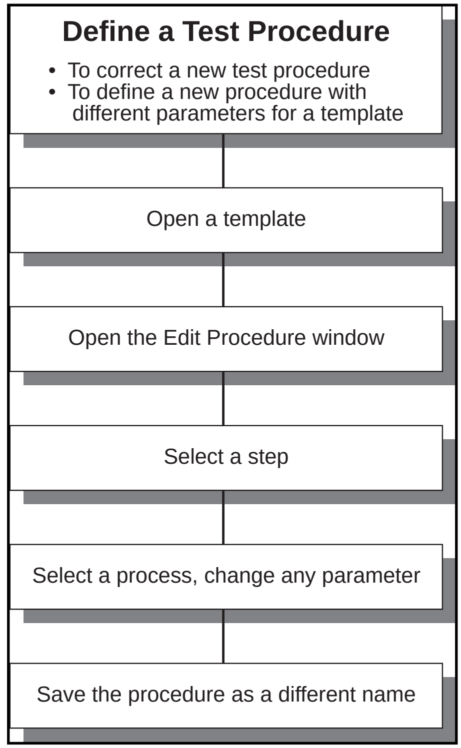
Note *Before you begin to define additional test procedures for a test template, be sure to run the first procedure you define to determine if the template and procedure operate as expected.*

Each test template can include up to 999 test procedures associated with it. You may choose to change one or two parameters for each test procedure. *For example*, you may want a series of test procedures that change the amplitude (end level) of a process.

Continued…

Designing a Test

Edit Guidelines (continued)



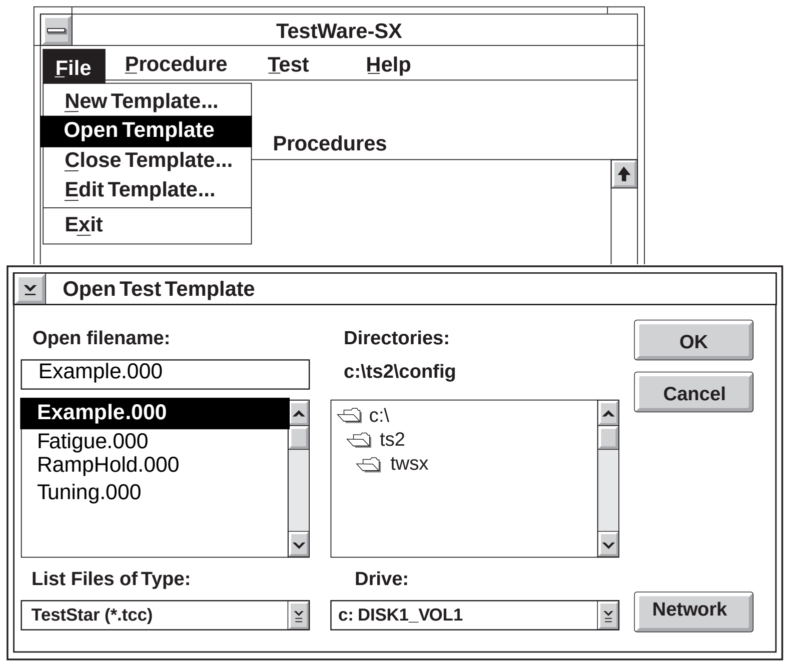
Step 1 Open a template

A Select the File menu in the TestWare-SX window.

B Select Open Template in the File menu.

C Select EXAMPLE.000 in the Files list in the Open Test Template window.

When you highlight a file, the file name is shown in the File entry field.



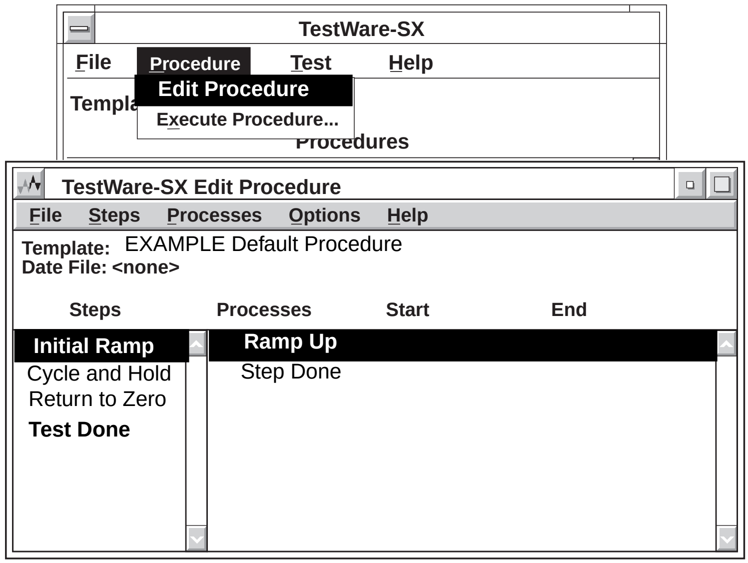
Step 2 Open the Edit Procedure window

The default procedure is actually the test template.

A Select the Procedure menu in the TestWare-SX window.

B Select Edit Procedure in the Procedure menu.

The Edit Procedure window shows the name of the test procedure, lists the steps of the procedure, and lists the processes of the selected step.



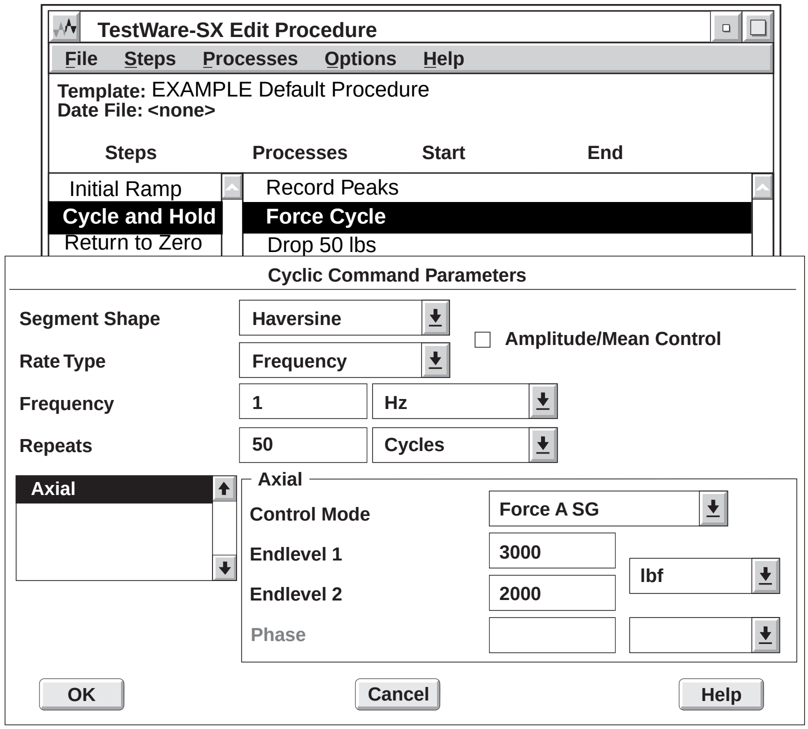
Step 3 Select the step and process to be edited

Repeat steps 3 and 4 for any process you wish to change

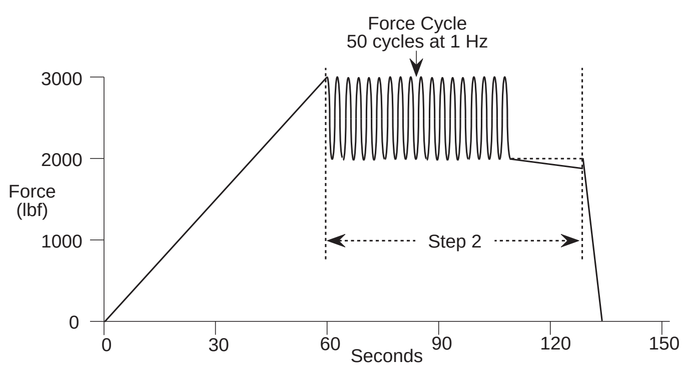
A Highlight the step in the Steps column you want to edit.

B Double-click the process you want to edit. This displays the procedure parameters window with the current settings shown.

Selecting the Force Cycle process in the second step of the EXAMPLE Default Procedure shows the settings for that process.



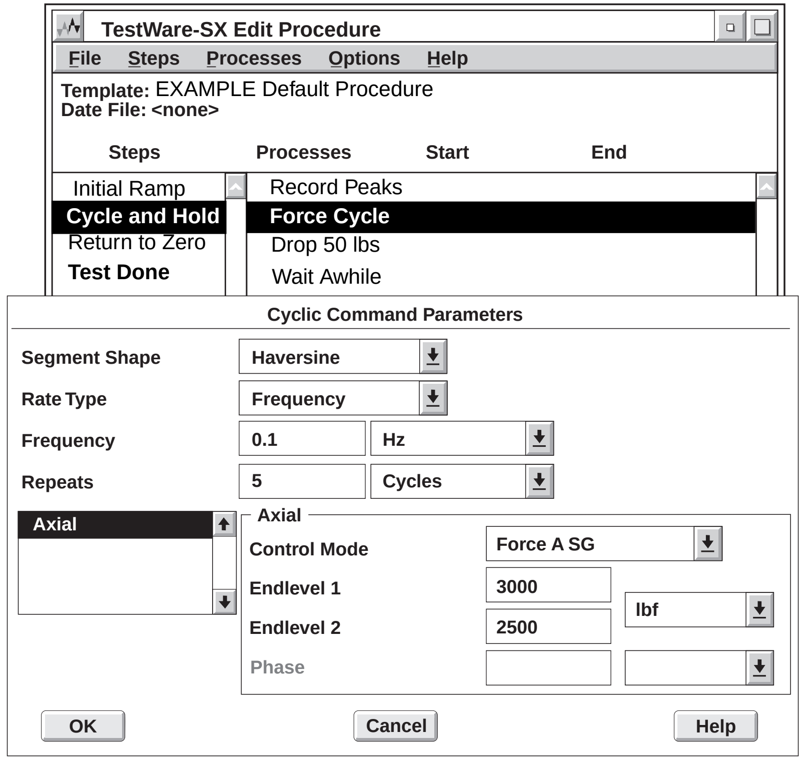
The current Force Cycle process is set for 50 cycles between 2000 and 3000 lbf at 1 Hz.



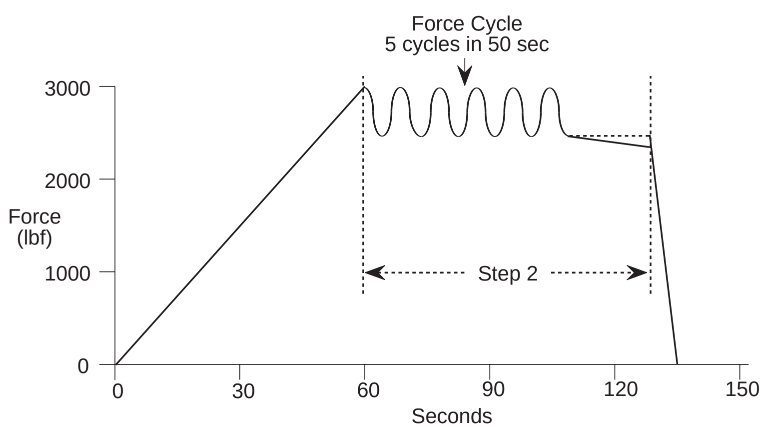
Step 4 Change the desired parameters

Change any parameters that you choose.

For example, changing the frequency from 1 Hz to 0.1 Hz allows 5 cycles over the same time period as the original procedure. Changing the amplitude (end level 1 from 2000 to 2500) creates a sine wave half as large as the original.



The new Force Cycle process is set for 5 cycles between 2500 and 3000 lbf at 0.1 Hz.



Step 5 Save the test procedure

A Select the File menu.

⚠ CAUTION

Do not select Save in the File menu to save a new procedure. Selecting Save replaces the procedure you opened with the new procedure.

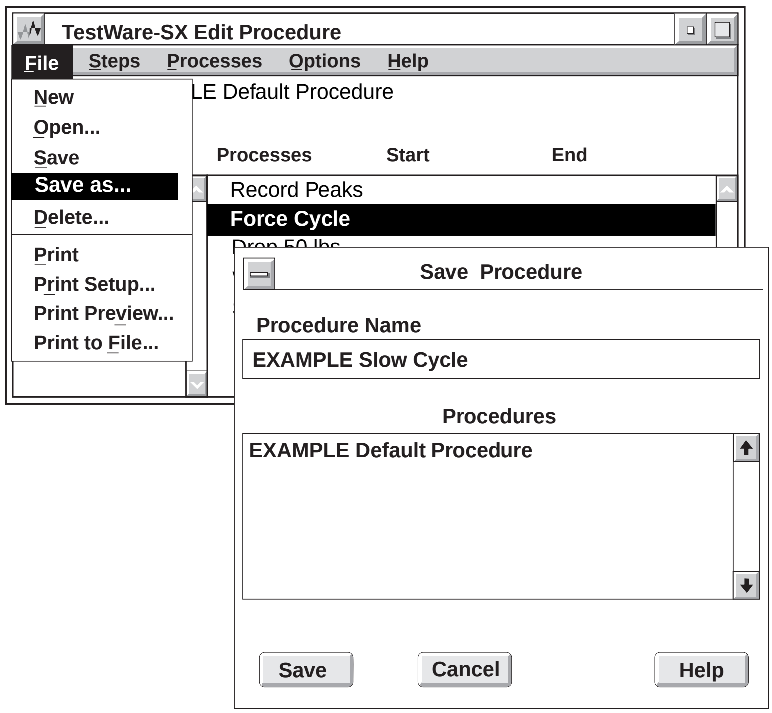
Use the "Save as" menu selection to save the procedure without replacing the procedure file that is opened.

B Select Save as in the File menu.

C Enter a name for the new version of the test procedure and press the Save pushbutton.

If you select Save, the EXAMPLE default procedure will be replaced with the new procedure.

Select Save as to define a new procedure for the test template.



Section C: Editing a Test Template

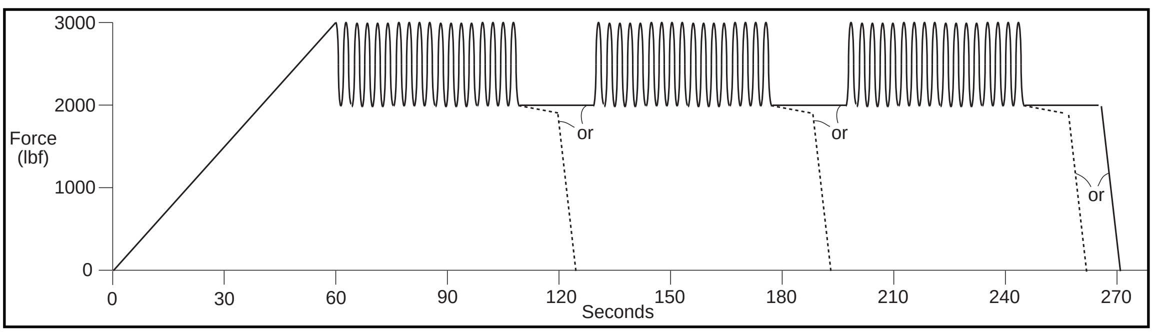
Procedure

Prerequisite

This section uses the test template created in Section A. This section also assumes you have opened the TestWare-SX application.

Edit example

The following procedure edits the example template created in Section A. A loop is added to repeat the Cycle and Hold step three times.



The new template will produce this waveform.

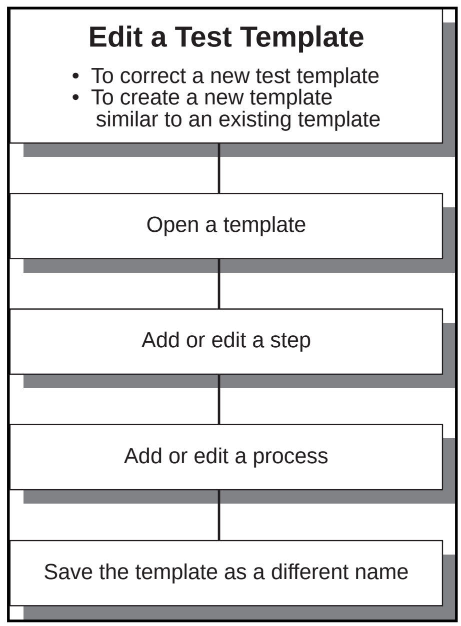
Edit Guidelines

There are two reasons you would use the edit template function:

- To correct a template you already defined.
- To create a new template.

Editing a test template allows you to do the following:

- Change the sequence of any step or process in a test template.
- Add new steps and processes.
- Change the triggers of any process.



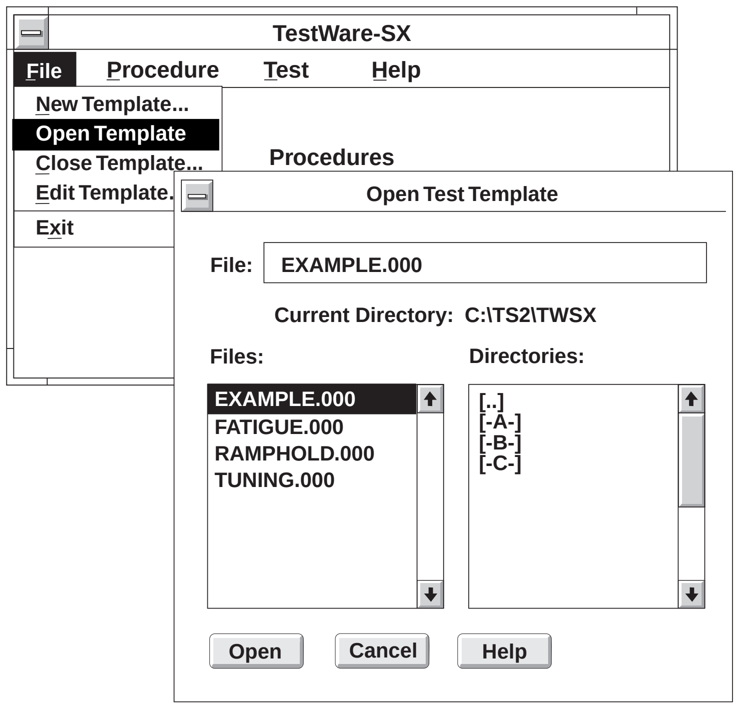
Step 1 Open a template

A Select the File menu in the TestWare-SX window.

B Select Open Template in the File menu.

C Select EXAMPLE.000 in the Files list in the Open Test Template window.

When you highlight a file, the file name is shown in the File entry field.

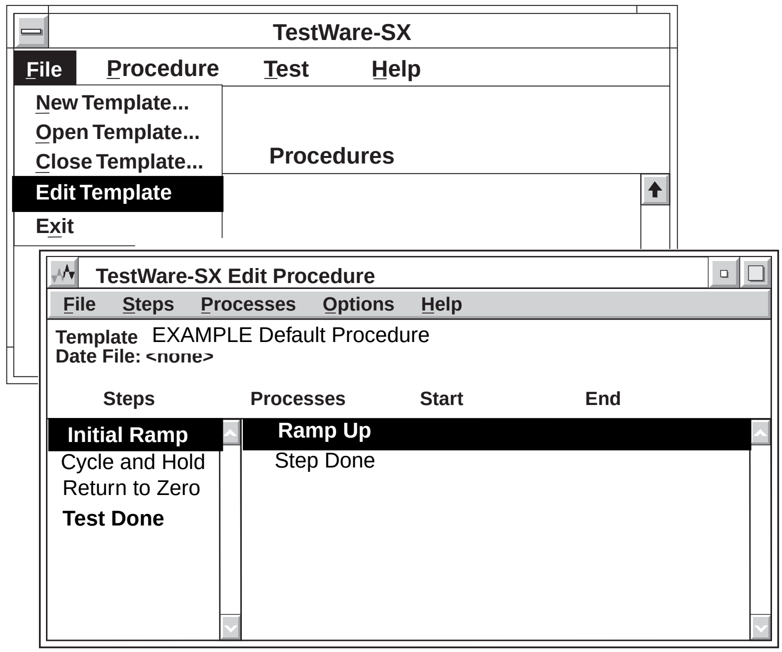


Step 2 Open the Edit Template window

A Select the File menu in the TestWare-SX window.

B Select Edit Template in the File menu to show the Edit Template window.

The Edit Template window shows the name of the EXAMPLE template, which we will change and rename.



Designing a Test

Step 3 Add a loop

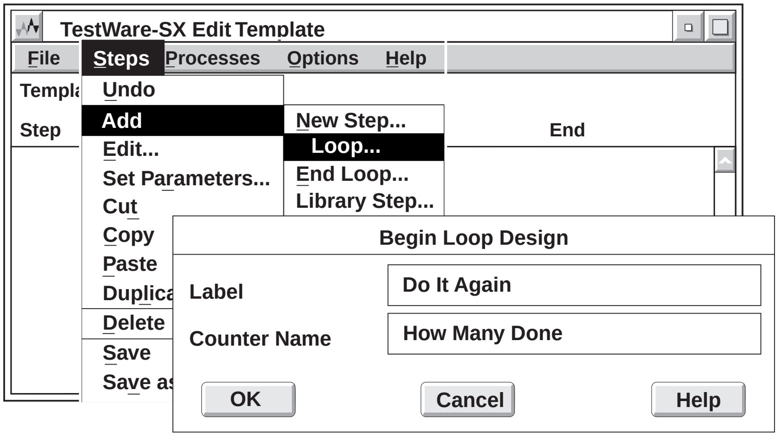
The steps following a loop step (and before the end loop designation) are repeated according to the number of selected repeats (entered in the next step).

A Select the Cycle and Hold step to insert the loop step before it.

B Select the Steps menu in the Edit Template window.

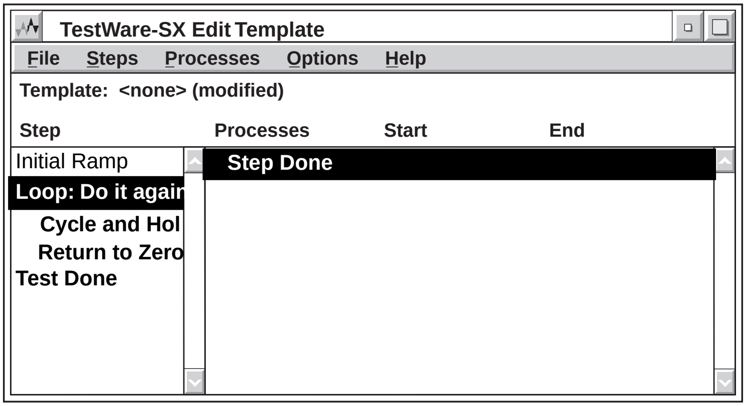
C Select Add in the Steps menu.

D Select Loop in the Add sub menu.



Enter a name you wish to call the loop step. In this example the step is called **Do It Again**. Also enter a name for the loop counter. In this example the loop counter is called **How Many Done**.

The steps that follow the loop step are indented which indicates they will be repeated. The indented steps are part of the loop.

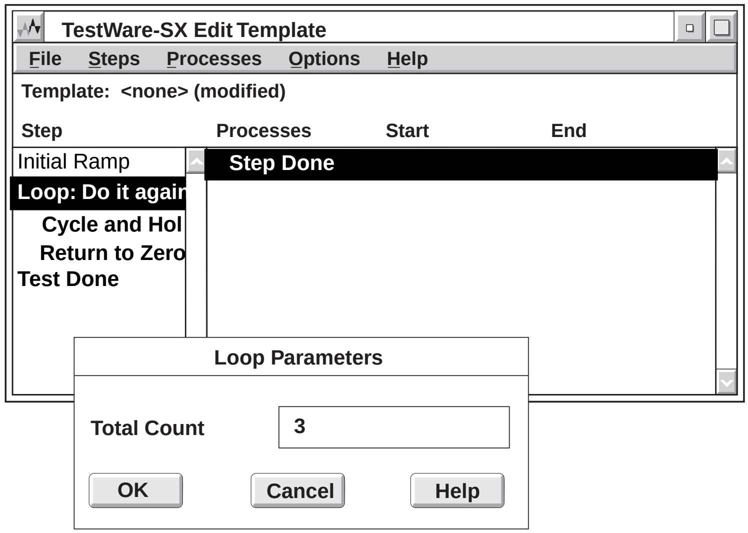


Step 4 Enter the number of loops

A Double-click the Loop: Do It Again step to display the Loop Parameters window.

B Enter the number of repeats.

The loop step is the only step you can double-click to reach another window.



Step 5 Add the end of the loop

This step adds the end of loop designation in the incorrect position to illustrate how to cut and paste a step later in this procedure.

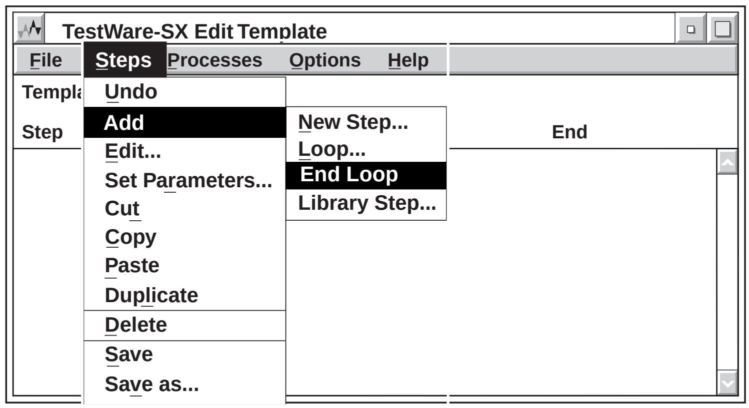
A Select Test Done in the Steps list.

B Select the Steps menu in the Edit Template window.

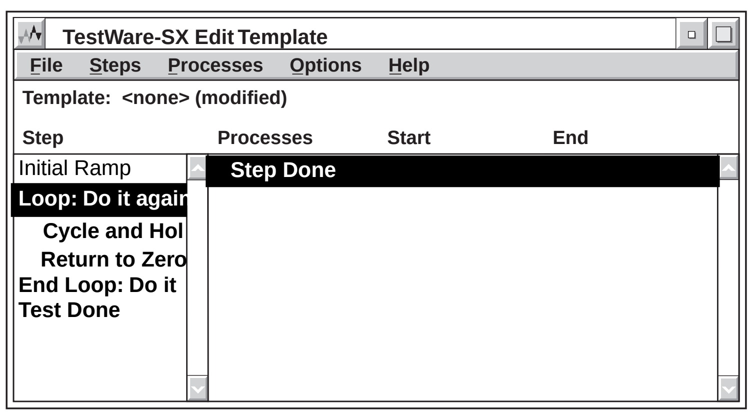
C Select Add in the Steps menu.

D Select End Loop in the Add sub menu.

A loop step requires an end loop designation.

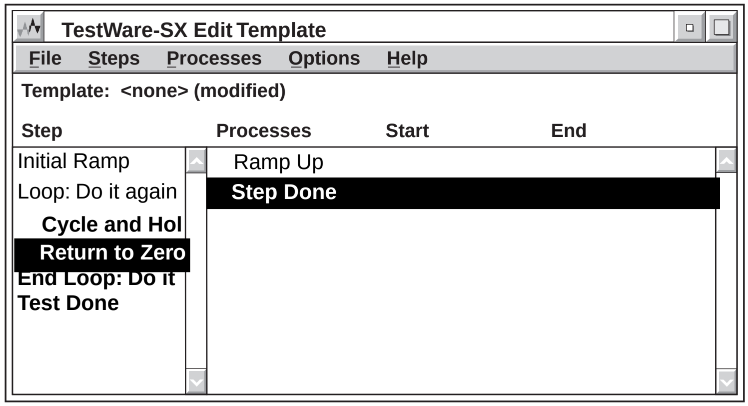


The loop consists of the Cycle and Hold step and the Return to Zero step.



Step 6 Cut and paste a step

We do not want to include the Return to Zero step in the loop. This step moves the Return to Zero step out of the loop.



A Select the Return to Zero Step in the Step list.

B Select the Steps menu.

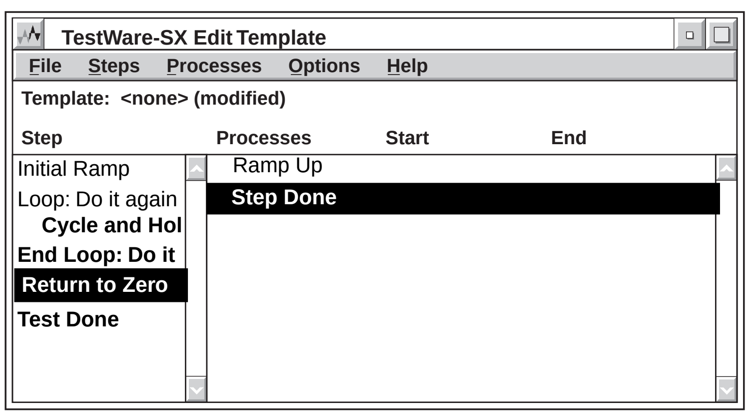
C Select Cut in the Steps menu. This removes the Return to Zero step from the list.

D Select Test Done in the Step list. We will paste the Return to Zero step before Test Done.

E Select the Steps menu.

F Select Paste in the Steps menu. This inserts the Return to Zero step back into the list.

The Return to Zero step is now after the loop.



Step 7 Save the template with a different name

⚠ CAUTION

Saving a template after editing it causes any procedures associated with it to be deleted.

Use the “Save as” menu selection with a different template name if you want to retain the original template and any test procedures.

A Select The File menu in the Edit Template window.

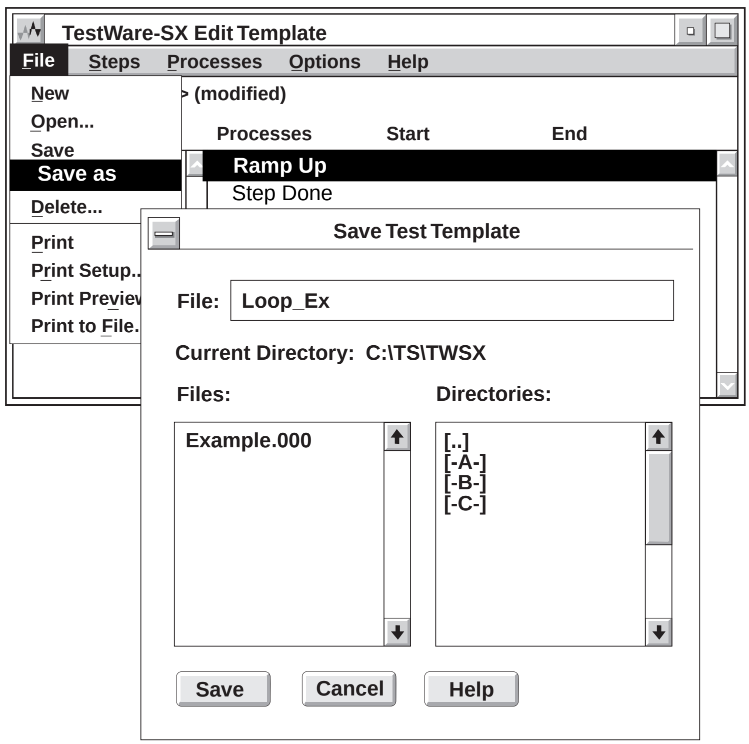
B Select Save as in the File menu.

C Type the name you wish to call the template in the File entry field and press the Save pushbutton.

The name of a test template file must be 8 letters or less with no spaces.

The example template that was edited in this section is called “LOOP_EX”.

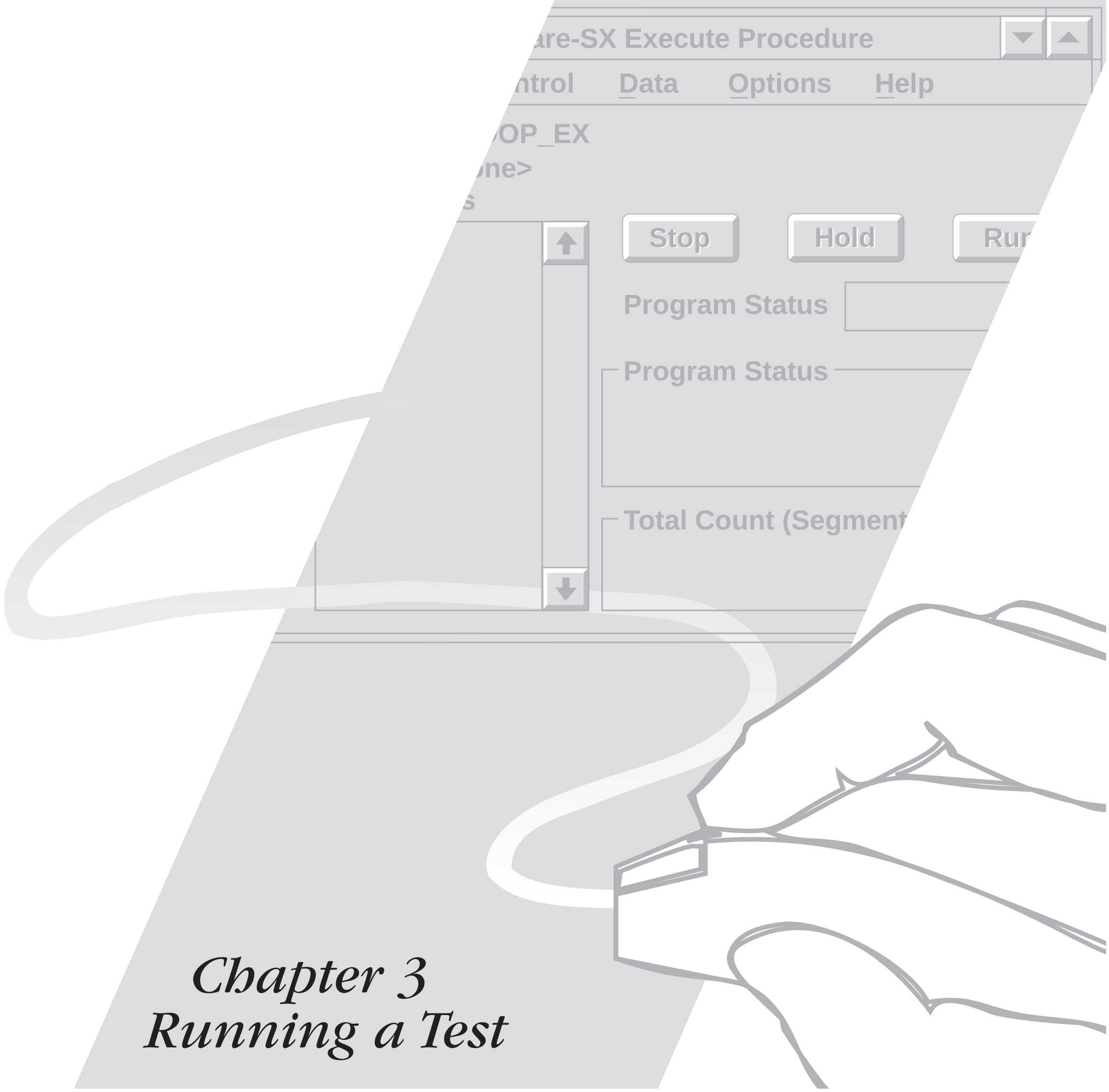
When the template is saved, the template name is shown in the Edit Template window.



MTS®

TestStar™ II

Control System



Chapter 3
Running a Test

Chapter 3

Using TestWare-SX

Note *The term '**test procedure'** as used within TestWare-SX represents a specific test program. It does not include all aspects of material testing.*

Contents

Prerequisites

TestWare-SX is just one part of your test strategy. You need to determine how to integrate TestWare-SX with TestStar and your specific hardware configuration to establish a sequence of operation.

For example, designing a test can occur anytime after the sensor signals and control channels are defined (TestStar Edit menu) But you can't actually run the test program until the following are true.

- A test procedure to control the test has been designed and saved. Designing a test procedure is described in Chapter 2.
- You should have a TestStar configuration file or have all TestStar controls set for the specific test you want to run.

Note *A TestStar configuration file includes all settings for all the TestStar windows. A default configuration is automatically loaded when you log on. A different one can be loaded if you want. We recommend you have a configuration file for each type of test you run.*

- The system has been completely set up and calibrated. This is described in the TestStar Installation manual.
- The test specimen is installed and any hardware appropriate for the test is installed and ready.

Also, during and after a test you will probably need to interface with other equipment.

Section A: Opening a Test Procedure

This procedure starts TestWare-SX and opens a test procedure for editing or execution.

Note *Refer to Appendix C for instructions on how to launch a test procedure with an icon.*

Prerequisites

You must have a predefined test procedure before you can open one. *Section A: Creating a Test Template* on page 84.

Procedure

The following steps apply only if you are using a data file.

Step 1 Start the TestStar application

TestStar must be started before TestWare-SX.

A From the desktop, open the TestStar folder and double-click the TestStar icon.

B From the File menu, select Open. This displays the Open Configuration window.

C Select a TestStar configuration file that is compatible with the test procedure to be selected in the next step.

Step 2 Start the TestWare-SX application

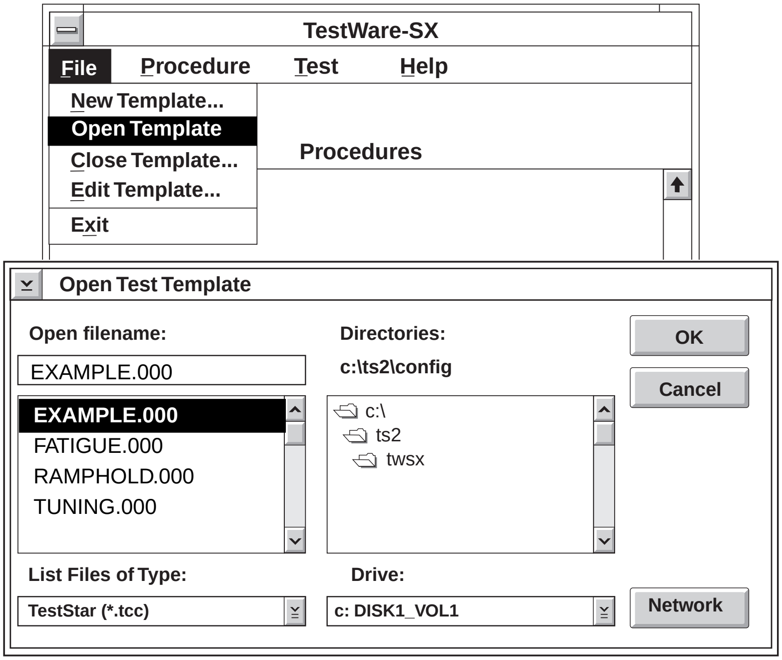
From the desktop, in the TestStar folder, double-click the TestWare-SX application program.

Step 3 Select a test template

A From the TestWare-SX File menu, select Open Template.

B While in the Open Test Template window, select the template needed for your test procedure.

- Remember that a template may have many procedures. At this time, you are still selecting only the "parent" test.



Step 4 Open a test procedure

To run a procedure

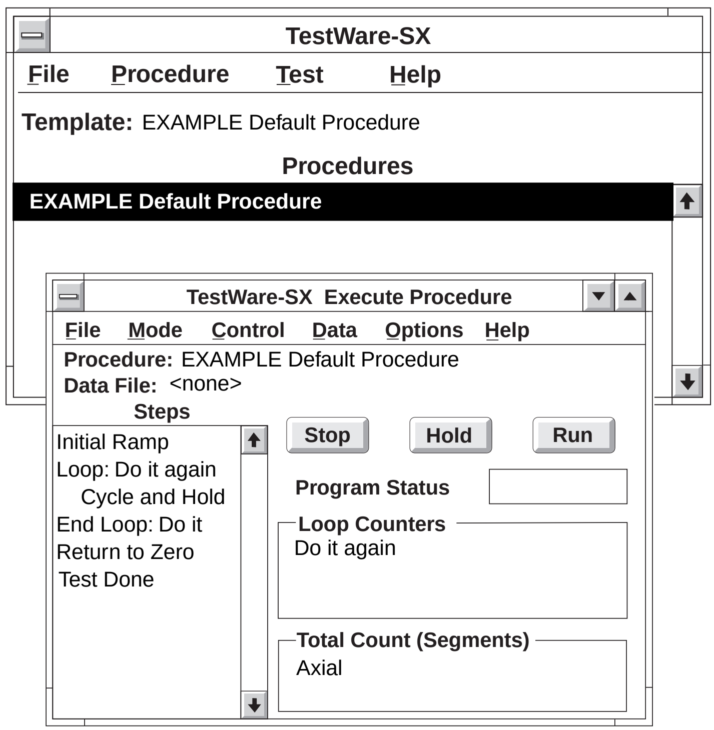
Double-click the specific test procedure to open it in the Execute Procedure mode. In this case the LOOP_EX procedure is opened.

Note *If the procedure has a data file associated with it, the* ***Existing Data File*** *window opens. For normal operation select* ***Neither*** *to keep the current status. Also, select if you want to Append or Overwrite the data file.*

To edit a procedure

Highlight the specific test procedure and use the File menu to open the procedure in the Edit Procedure mode. *Section B: Editing a Test Procedure* on page 115.

Once a procedure is opened, use the mode menu to enter the edit mode or the execute mode.



Step 5 Select the data file format

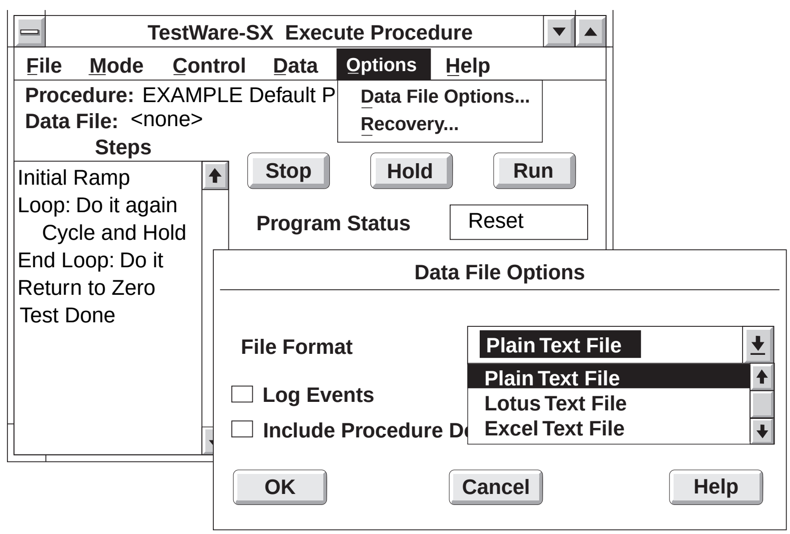
If your test is not gathering data or you completed the Existing Data File window, go to Task 7.

The data file format may have been established with the test template. Complete this step to check or change the data file format.

From the Execute Procedure window, select the Options menu, then select Data File Format.

- Enable Log Events if you want to log when the Run, Hold, and Stop pushbuttons are pressed. This also records when interlocks become active.
- Enable Include Procedure Description if you want a listing of the test procedure steps, processes, and parameters included with the test data. The procedure description is the same information that can be acquired with the Print Preview function in the File menu.

Select one of the three Data File Formats that is best suited for the application you intend to use with the test data.

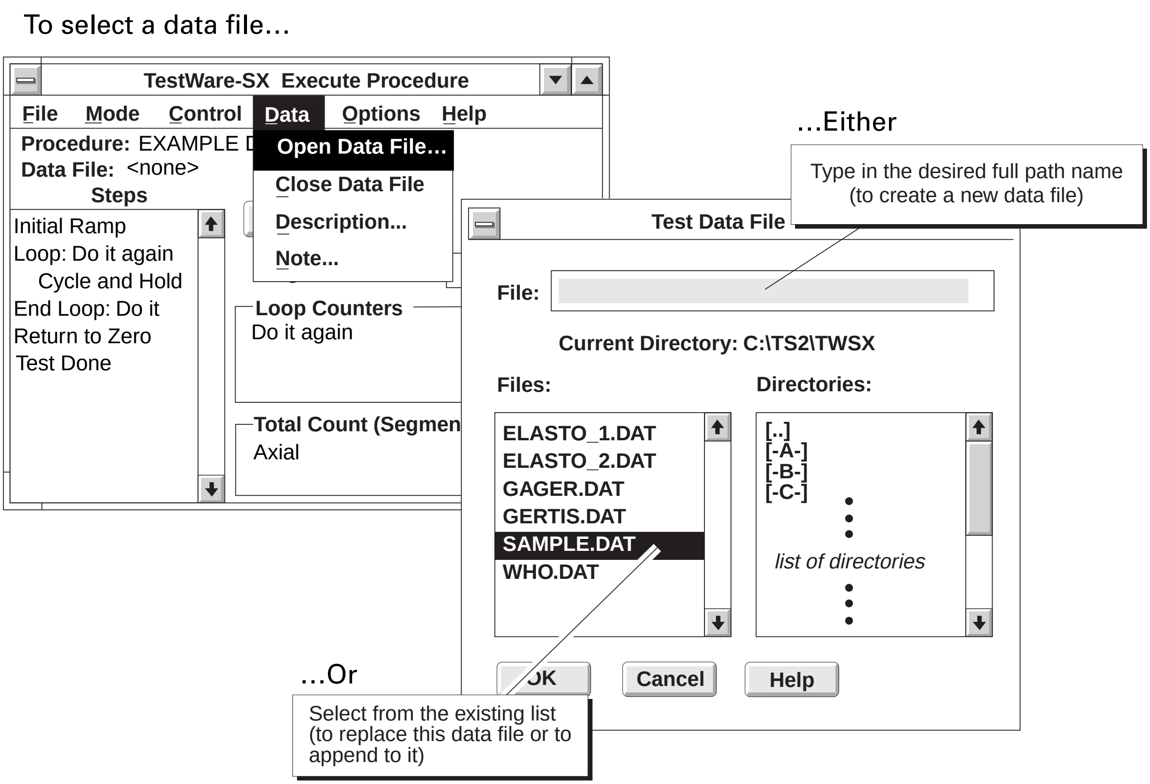


Step 6 Open the Test Data File window

If your test is not gathering data or you completed the Existing Data File window, go to Step 7.

From the Execute Procedure window, select Data, then select Open Data File.

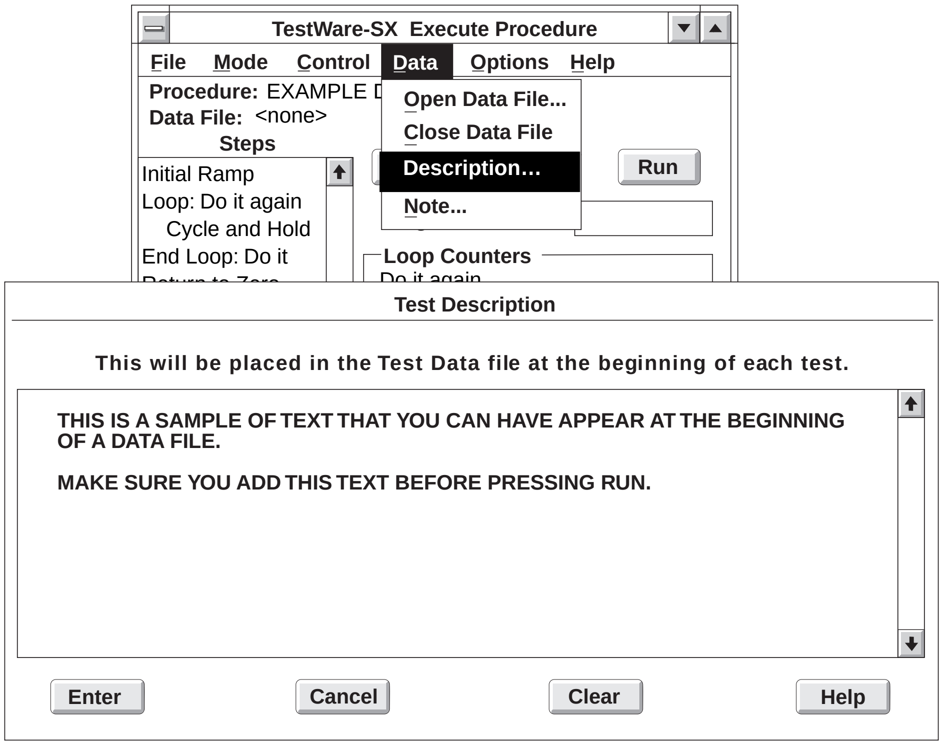
You may either select an existing file, or else create a new one. If you create a new file, its name cannot exceed 8 characters (not including the .DAT extension).



If you select an existing data file, the system displays the Existing Data File window where you can restore the last test procedure that used the data file. You can also select to append or overwrite the data file. Select the restore option Neither to retain the procedure you opened in step 3.

Step 7 Enter the test description

If you want a description entered at the beginning of the data file, choose Description from the Data menu and enter your text. Make sure to do this *before* you start the test.



Step 8 Set up the auto save feature

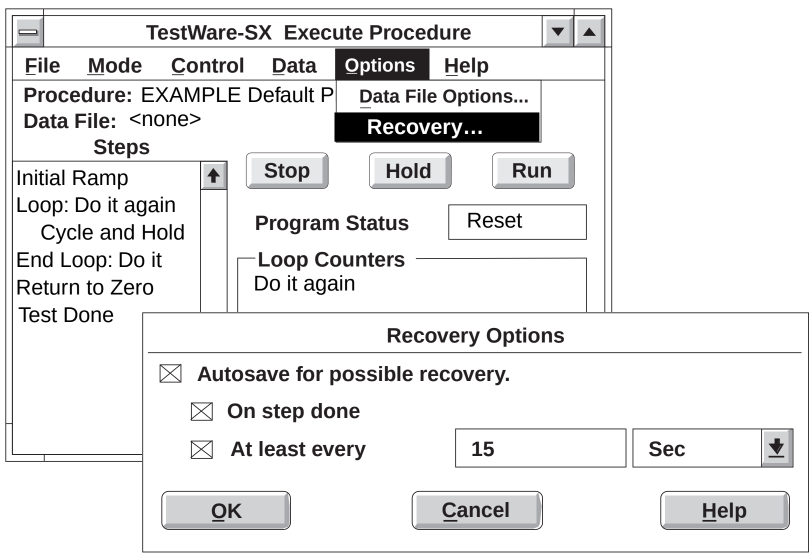
This is an optional step.

Setup the recovery options if you want to protect your data (and test status) from a power failure or if you have a reason to suspend the test then continue it. Depending on the recovery selections, you can restart your test from the point it was stopped.

- **Auto save for possible recovery**—saves the test status any time the program status changes from run to stop or stop to run (including interlocks). Along with this selection, the following selections can also be included.
- **On step done**—saves the test status at the end of every step.
- **At least every**—saves the test status periodically. Define how often a snap shot of the test status occurs. Type a value in the entry field and select the units with the list icon.

The default configuration enables the Auto save for possible recovery selection.

This saves the test status when the Run, Hold or Stop pushbutton is pressed.



Section B: Running a Test Procedure

Prerequisites

You must have a test procedure open and the execute mode selected. You also must comply with the prerequisites listed at the beginning of this chapter.

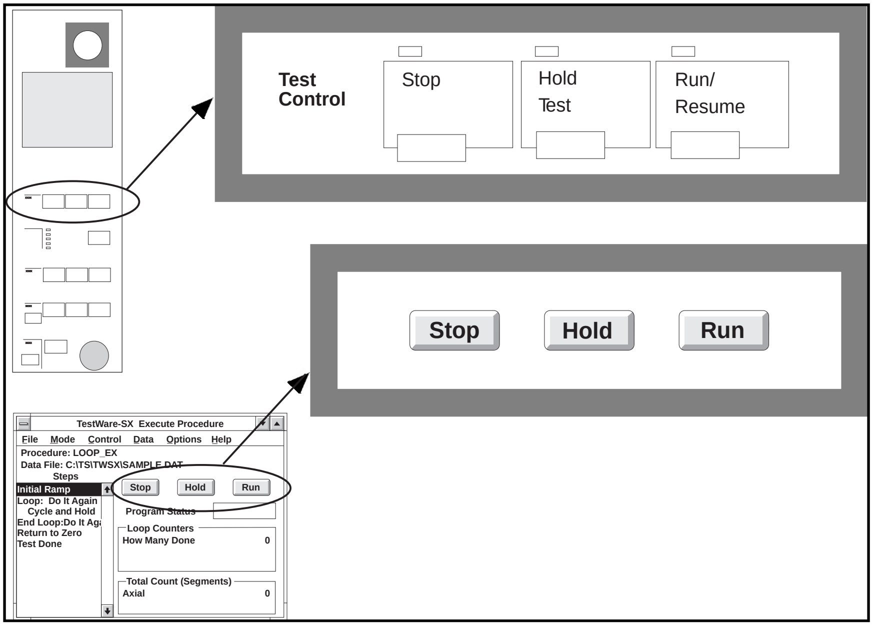
Procedure

Controls overview

You can control the test by using the switches on the load unit control panel…

or,

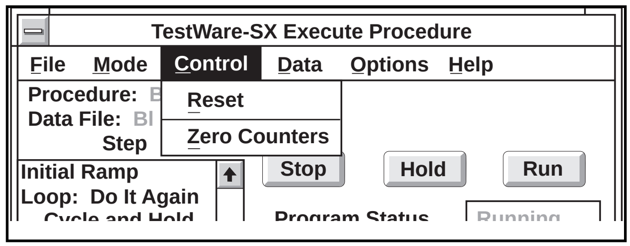
…by using the pushbuttons in the Execute Procedure window.



Step 1 If necessary, reset the counters

This step is not necessary if this is your first test since starting TestWare-SX. If this is a repeat procedure (either a new specimen or a restart after you have stopped the test), then you will want to reset the counters.

- Reset clears the counters and initializes the test procedure to run the test from the start.
- Zero Counters clears the Total Counters indication.



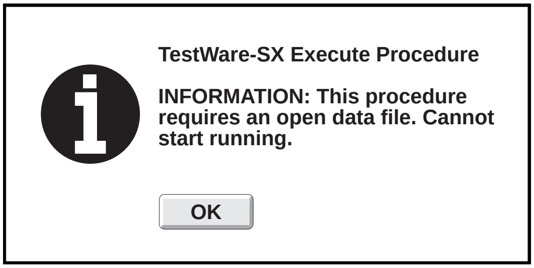
Step 2 Start the test

Note *It is a good idea to print your configuration file and test procedures. Printed copies of these files help you determine the exact setup of the TestStar system in the unlikely event you need it.*

A Arrange windows on the screen to meet your personal test requirements.

B With hydraulic pressure on high, start the test by pressing the **Run** pushbutton either on the LUC or in the Execute Procedure window.

If you get the dialog box as shown when you start the test, you have not opened a data file.



Step 3 While the test is running

You can have several options available to you while a test is running. Each option depends on the types of processes you used to create the test procedure and what occurs during a test. Besides using the Run, Stop, and Hold pushbuttons, some of the following may be available during your test:

- Adding notes during a test
- Monitoring sensor signals
- Compensating a file playback process
- Using an operator event
- Clearing an interlock
- Recovering a test
- Using special processes

Adding notes during a test

You can add operator notes to the data file whenever the test is stopped or on hold. This could, for example, include comments on changing test conditions or the reason why the test was interrupted. In this case, the note is placed at the data file's current location (between the last data recorded and the next data to be recorded) instead of at the end.

See ***Add operator's note to the end of the data file*** on page 149 for instructions on how to add a note.

Monitoring sensor signals

You can monitor up to four meters and/or a scope display.

Each meter can be set up to monitor any sensor input signal and the display can be configured to show:

- timed data
- peak/valley data
- span/mean data
- min./max. data

The scope can be set up to show a sensor signal like an oscilloscope.

See Chapters 3 and 5 in the Reference manual for procedures to use the meters and scope.

Compensating a File Playback process

If you have a test that contains a file playback process and the process has manual compensation selected, you will see the file playback compensation window when the process begins. The file playback compensation window adjusts the mean level or amplitude of the playback command.

For example, use this window if your program is not reaching its intended peaks.

See ***File Playback Compensation Window (manual)*** on page 301 for more information.

Using an operator event

If your test contains an operator event process, you will see the Operator Event window when the process begins. This window can include up to three pushbuttons representing three different processes. Each pushbutton includes a description of its function. Each process can trigger another process.

For example, one operator event may trigger a short data acquisition process each time the pushbutton is pressed. Another operator event may suspend the test until you press the pushbutton to acknowledge that you performed the task.

See ***Operator Event*** on page 320 for more information.

Clearing an interlock

A test can stop running if an interlock occurs. Interlocks monitor specific conditions. A test can have from none to many interlocks enabled.

For example, some tests are run until the specimen breaks. This type of test usually has limit detectors configured to stop the test and turn off hydraulic pressure.

See Chapter 1 in the Reference manual for instructions to identify and correct an interlock.

Recovering a test

You may intentionally stop your test to conclude it later, or your test may stop for other reasons. In both cases, the test can be restarted from the point where it stopped.

See ***Section C: Recovering a Test Procedure*** on page 150 to recover a test procedure.

Using special processes

Some special processes present run-time windows that provide status unique to optional TestWare applications.

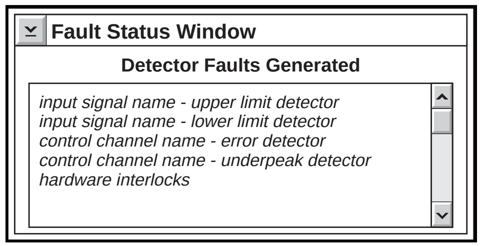
See the appropriate manual for information about run-time capabilities.

Step 4 When the test ends

The test can end due to a number of reasons.

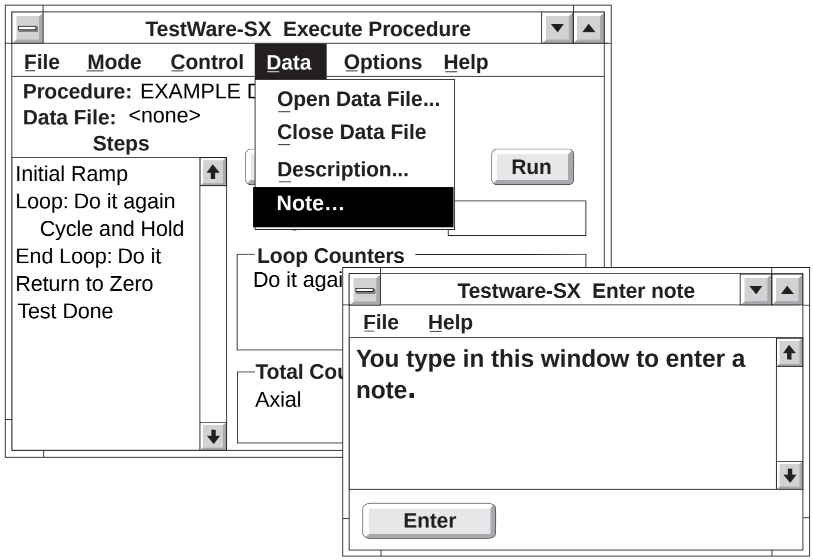
- You pressed the Stop pushbutton.
- The test procedure is done.
- A TestStar detector (limit, error, or underpeak) triggers and stops the test. Detectors are usually set to stop the test when they sense conditions that indicate the specimen is about to fail, or conditions that you consider to indicate the test is complete. *See Chapter 1 in the TestStar Reference manual to clear detector faults and interlocks.*

If a detector activates, this windows appears to identify the reason the test stopped.



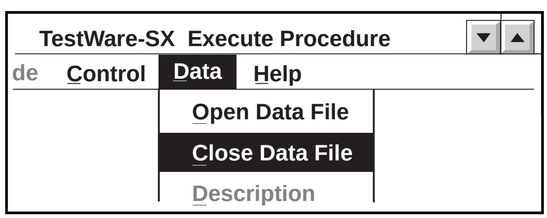
Step 5 Add operator's note to the end of the data file

This step is optional. When the test is done, you can add a note to be appended at the end of the data file. Use the Data menu and select Note. Type the note you want and press the Enter button.



Step 6 Close the data file

Close the data file from the Data menu.



Section C: Recovering a Test Procedure

This is optional. It is only needed if your test stopped before it was considered done.

Prerequisite

You must have set up the auto save feature.

Procedure

Step 1 Determine the reason to restart the test

There are three basic reasons that require you to recover a test. Note the reason you are recovering a test procedure.

- You wish to use a specific test procedure and data file for repetitive testing.
- The test stopped under controlled circumstances. *For example*, you chose to stop the test and continue it later.
- The test stopped because of an uncontrolled event. *For example*, a power failure shuts down the system causing the test to stop.
- You made a change to the test procedure and ran the test without saving the new procedure. *For example*, you edited the test procedure to optimize a parameter and wish to recover the new procedure.

Step 2 If necessary, restore everything the way it was

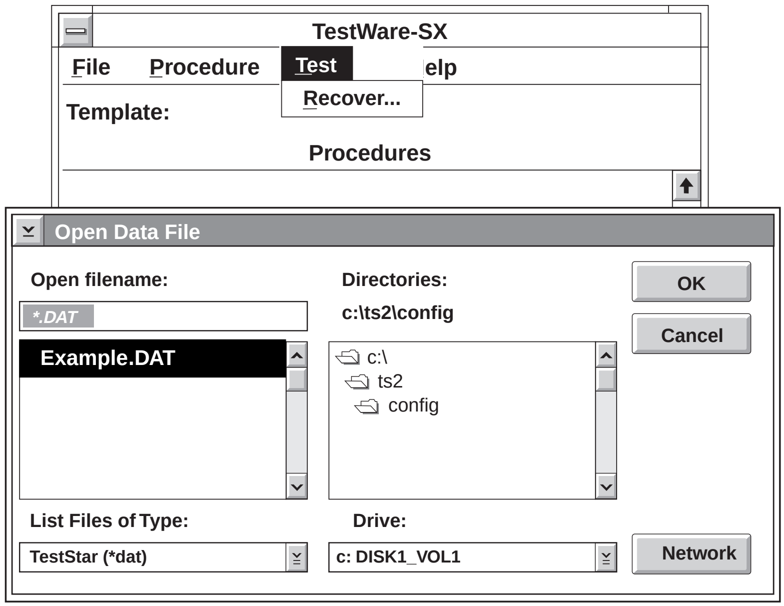
It may be necessary to log back onto TestStar or reinstall the specimen.

- If you have printed copies of your configuration file, controller definition file and the test procedure, use them to re-establish everything the way it was before the test stopped.
- Make sure you have the correct controller definition file loaded. This also means that the hardware associated with the controller definition must be connected.
- Get the configuration file back the way it was. You may have tweaked the tuning or optimized a detector. If you did not save the adjusted configuration file you will need to remember what you did.
- If necessary, start the TestWare-SX application.

Step 3 Select the data file to recover

A From the main TestWare-SX window, select Recover in the Test menu to display the Open Data File window.

B Select the data file name associated with the test you wish to restore and open it.



Step 4 Select the recovery options

Use the Existing Data File window to choose how to restore the test status and whether to overwrite or append the existing data file. Your choices depend on why you are recovering the test (step 1).

Note *If you edit the test procedure (and don't save it), you will see a dialog box that requests you to choose the original procedure or the modified procedure used during the test.*

*Select **Yes** to load the modified procedure that you edited for the test. The test status also saves changes to the procedure.*

*Select **No** to load the original procedure without any changes.*

For more information about test recovery, see Chapter 1, section D to find out how it works.

- Restore the procedure and test status when you want to restore a test from the point where it was stopped.
- Restore the procedure only if you want to run the procedure with a new specimen or if you edited the procedure for the test and did not save the revised procedure.
- Append data to the end of the file when recovering from a stopped test or iterations of a single test.
- Overwrite data to reuse the data file when old data is no longer needed.

Continued...

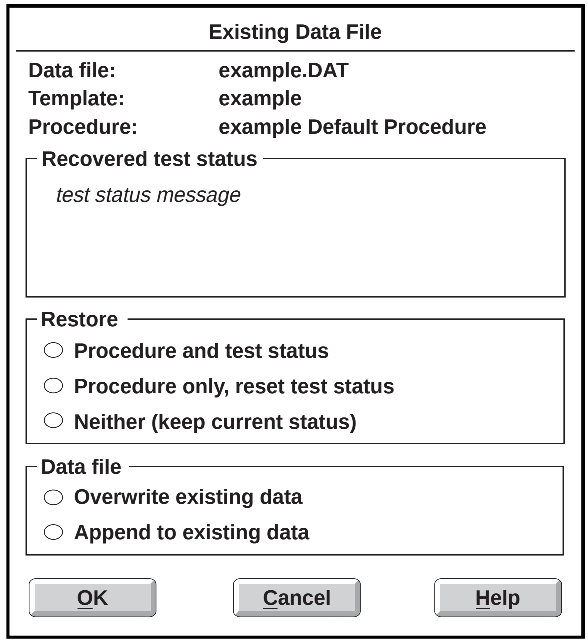
Step 4 Select the recovery options *(...continued)*

The test status message area displays the status of the test when the last test status was recorded.

For example, the test status would be done or reset if the test had been completed.

The test status would be stopped or running if the test was not completed.

A test status of abnormal means that the test was not stopped in the normal way (using the system menu icon).

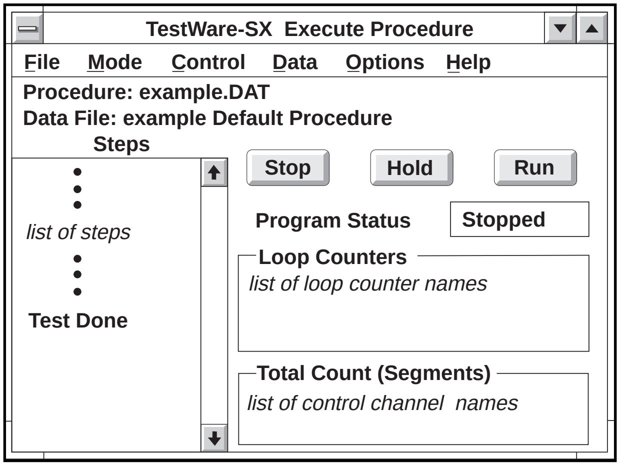


Step 5 Start the test

When the test procedure and data file are restored, the Execute Procedure window opens.

Press the Run pushbutton to start (or restart) the test.

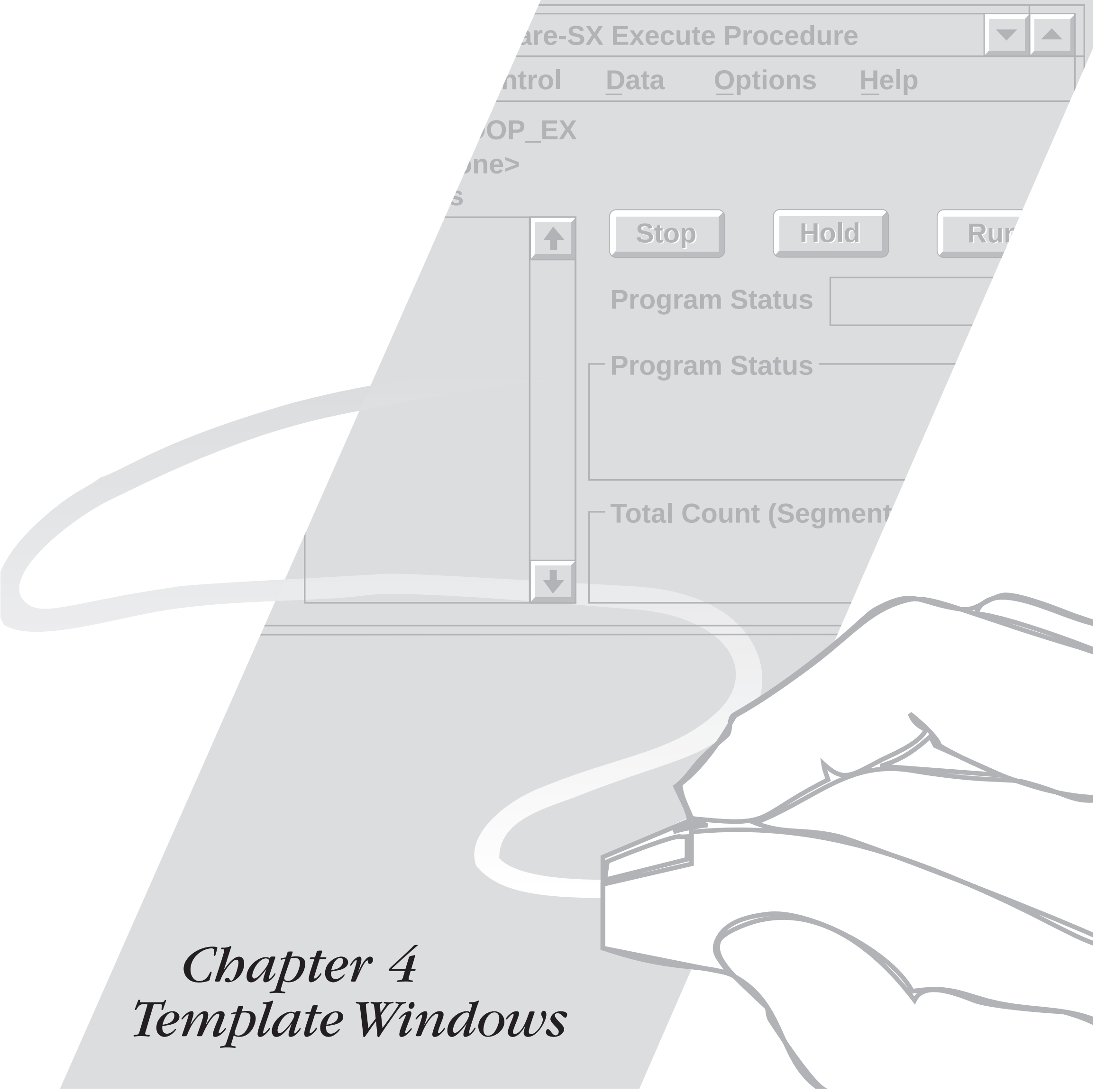
Note *If the test status is Done, you need to reset the test using the Control menu.*



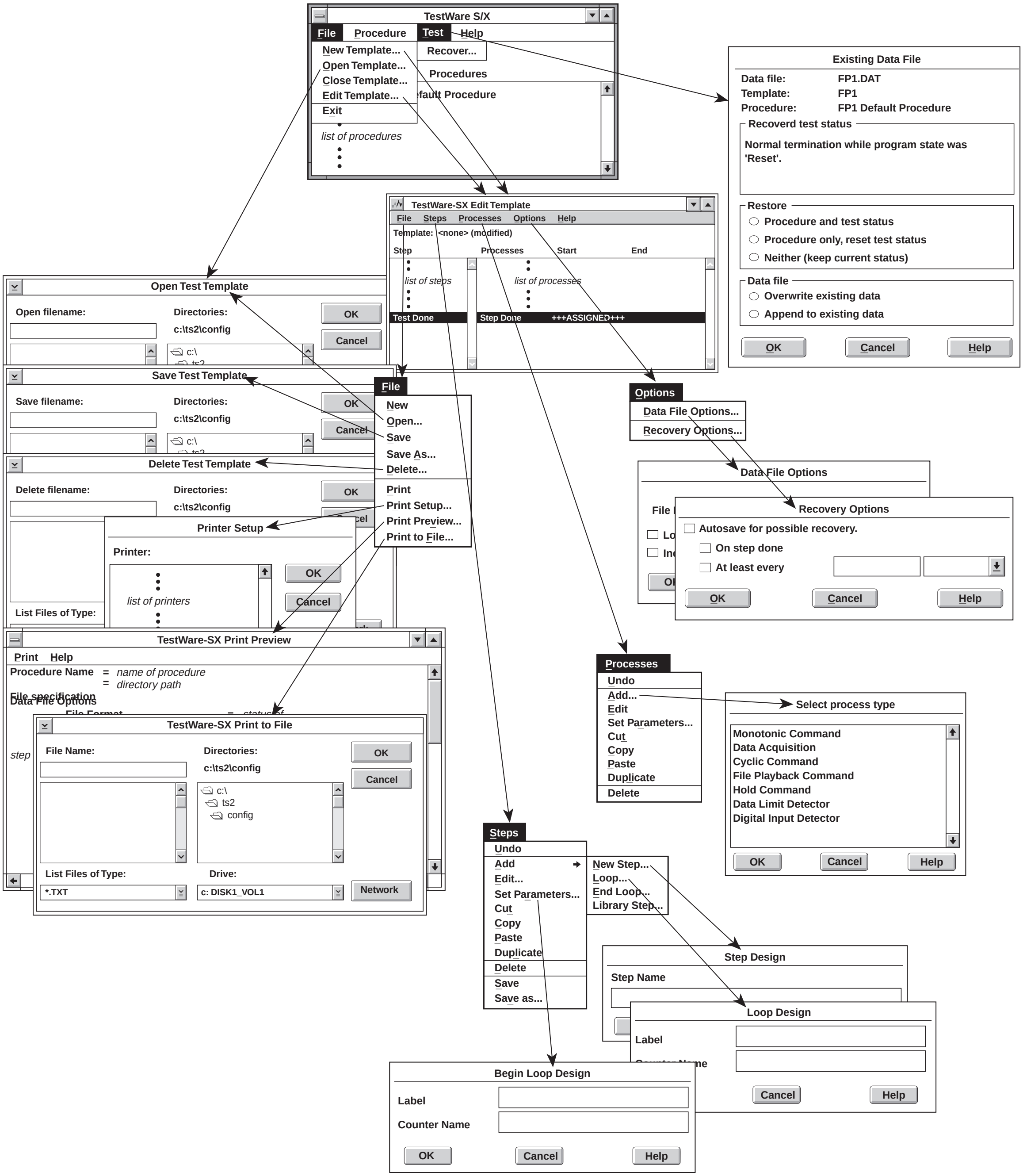
MTS

TestStar™ II

Control System



Chapter 4
Template Windows



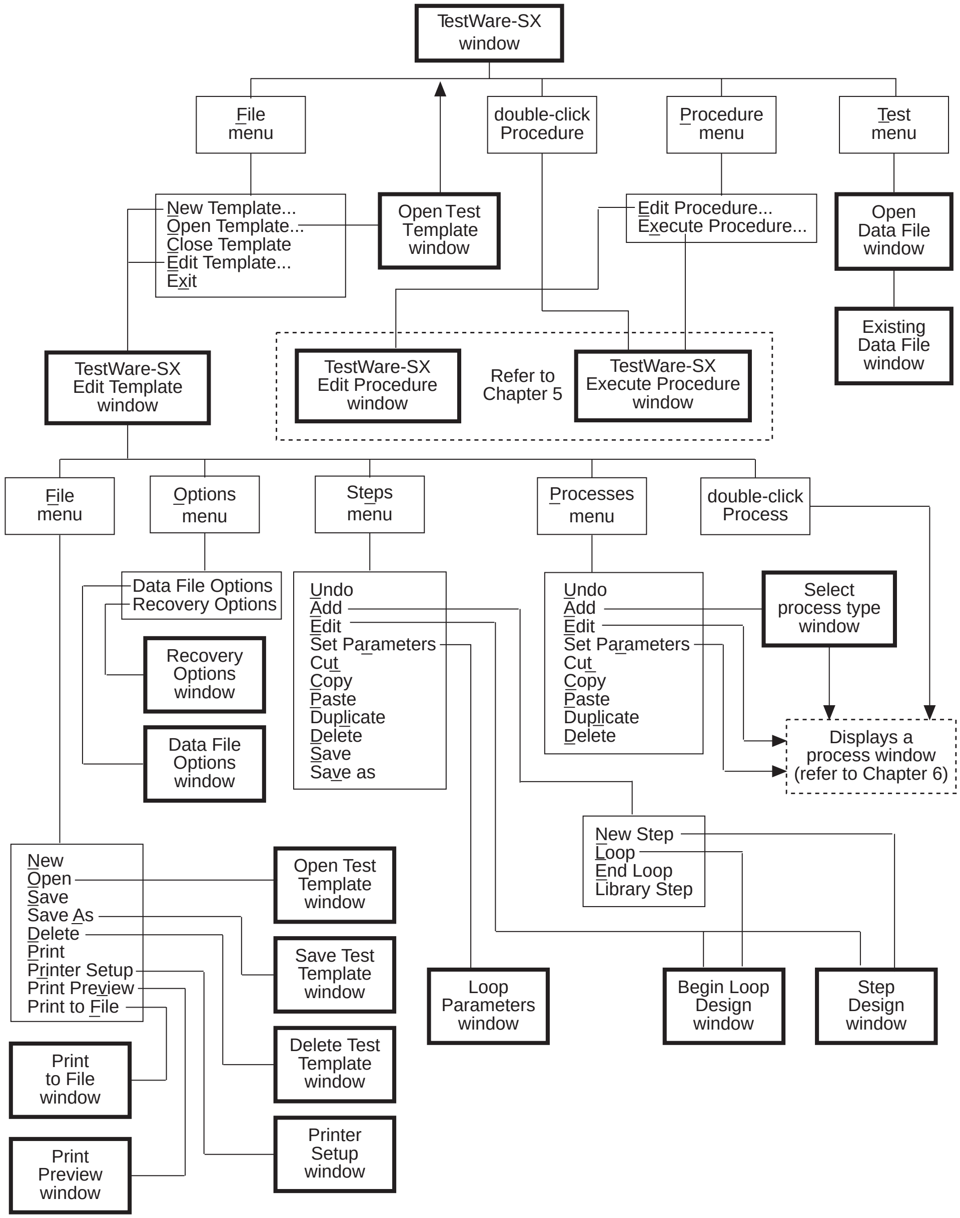
TestWare S/X
File
Procedure
Test
Help
New Template...
Open Template...
Close Template...
Edit Template...
Exit
Recover...
Procedures
Default Procedure
list of procedures
Existing Data File
Data file: FP1.DAT
Template: FP1
Procedure: FP1 Default Procedure
Recoverd test status
Normal termination while program state was 'Reset'.
Restore
Procedure and test status
Procedure only, reset test status
Neither (keep current status)
Data file
Overwrite existing data
Append to existing data
OK
Cancel
Help
TestWare-SX Edit Template
File
Steps
Processes
Options
Help
Template: <none> (modified)
Step
Processes
Start
End
list of steps
list of processes
Test Done
Step Done
+++ASSIGNED+++
Open Test Template
Open filename:
Directories:
c:\ts2\config
OK
Cancel
Save Test Template
Save filename:
Delete Test Template
Delete filename:
File
New
Open...
Save
Save As...
Delete...
Print
Print Setup...
Print Preview...
Print to File...
Printer Setup
Printer:
list of printers
List Files of Type:
Options
Data File Options...
Recovery Options...
Data File Options
Recovery Options
Autosave for possible recovery.
On step done
At least every
TestWare-SX Print Preview
Print
Help
Procedure Name = name of procedure
= directory path
File specification
Data File Options
step
TestWare-SX Print to File
File Name:
Directories:
c:\ts2\config
c:\
ts2
config
List Files of Type:
*.TXT
Drive:
c: DISK1_VOL1
Network
Processes
Undo
Add...
Edit
Set Parameters...
Cut
Copy
Paste
Duplicate
Delete
Select process type
Monotonic Command
Data Acquisition
Cyclic Command
File Playback Command
Hold Command
Data Limit Detector
Digital Input Detector
Steps
Undo
Add
Edit...
Set Parameters...
Cut
Copy
Paste
Duplicate
Delete
Save
Save as...
New Step...
Loop...
End Loop...
Library Step...
Step Design
Step Name
Loop Design
Label
Counter Name
Begin Loop Design
Label
Counter Name
OK
Cancel
Help

Chapter 4

Template Windows

This section describes the windows and menus related to templates. Some descriptions include references to topics and their chapter numbers where more information can be found.

Contents

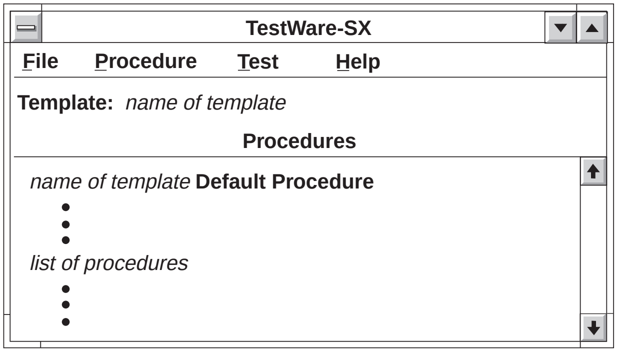


This diagram shows the relationships of the windows and menus of the template function. The darker boxes represent windows and the lighter boxes represent selections.

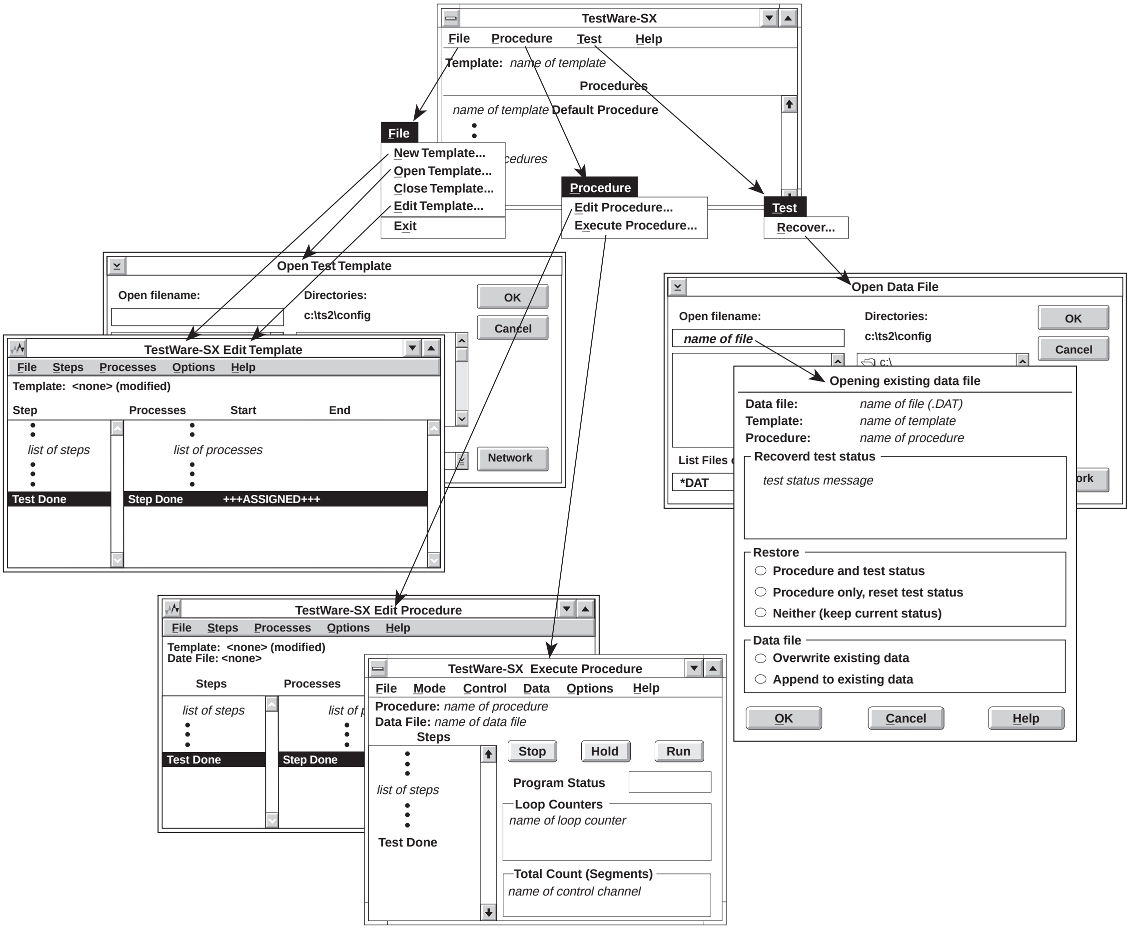
TestWare-SX Window

The TestWare-SX window is the first window you see when you open the application.

Use this window to access the template, procedure, and execution functions.



CONTROL	FUNCTION
File Menu	Allows you to create, open, close and edit test templates.
Procedure Menu	Accesses the Edit Procedure and Execute Procedure windows after you open or edit a template.
Test Menu	Displays the Open Data File window where you select a data file. If the data file has been previously used, it includes test status information that opens the Existing Data File window. This is where you can recover a test.
Template	Displays the name of the test template that is open. If no template is open, or you have not saved a new template, <none> is displayed.
Procedures	Displays a list of template procedures associated with the open test template. All templates include a default test procedure (which is actually the template). Template procedures are created with the Edit Procedure window.

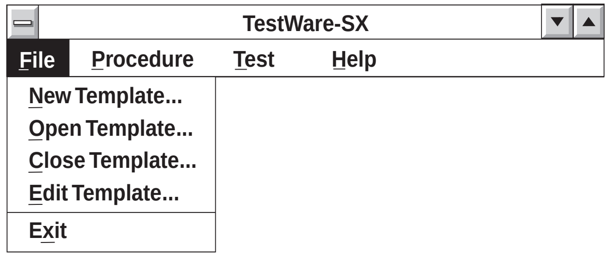


More information

Creating or editing a template	*Designing a Test* on page 83 *Edit Template Window* on page 171
Defining a procedure	*Edit Procedure Window* on page 199
Running a test	*Section B: Running a Test Procedure* on page 143 *Execute Procedure Window* on page 201

TestWare-SX File Menu

Use the File menu to open, create, or edit a template.



Menu selections

Menu	Function
New Template	Opens the Edit Template window with no steps or processes assigned.
Open Template	Opens a saved template and lists its associated procedures in the TestWare-SX window.
Close Template	Sets the current template back to <none>.
Edit Template	Opens the Edit Template window and displays the steps and processes assigned to the current template.
Exit	Leaves the TestWare-SX program and returns to the TestStar program.

Creating a new template

Selecting New Template displays the Edit Template window with no steps or processes listed. The Edit Template window allows you to create a new template by adding steps and processes.

Template Windows

Opening a template

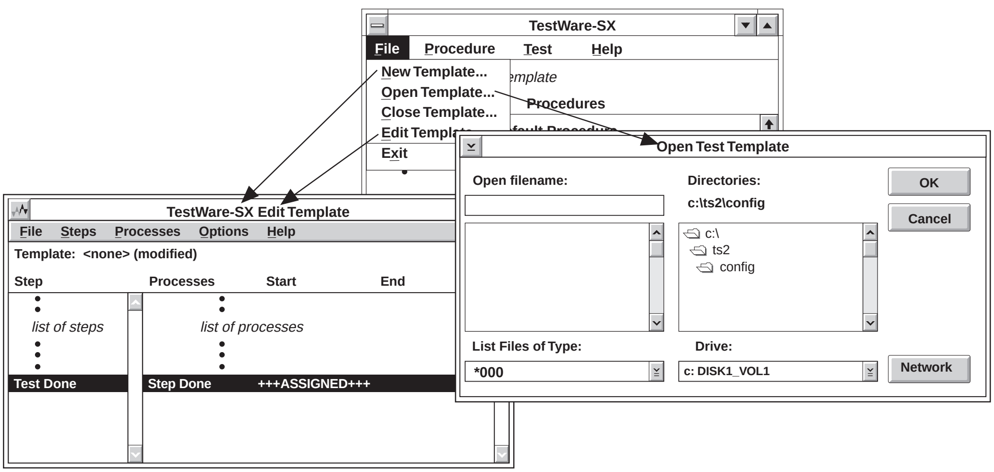
Selecting Open Template displays the Open Test Template window where you select the template you want to open. The name of the template you select and all the associated test procedures are shown in the TestWare-SX window.

Double-clicking one of the template procedures opens the Execute Procedure window where you can run the selected test procedure.

Editing a template

If a template is open, selecting Edit Template displays the Edit Template window. The Edit Template window displays the steps and processes of the open template.

If no template is open, selecting Edit Template displays the Edit Template window with no steps and processes (like the New Template menu selection).



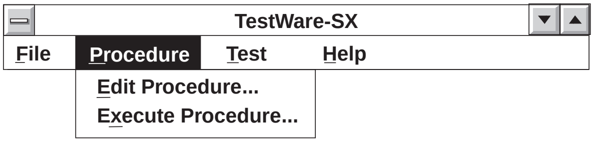
More information

Creating or editing a template — *Edit Template Window* on page 171
Section A: Creating a Test Template on page 84
Section C: Editing a Test Template on page 123

Opening a template — *Section A: Opening a Test Procedure* on page 136
Open Test Template Window on page 176

TestWare-SX Procedure Menu

Use the Procedure menu to move to the procedure functions.

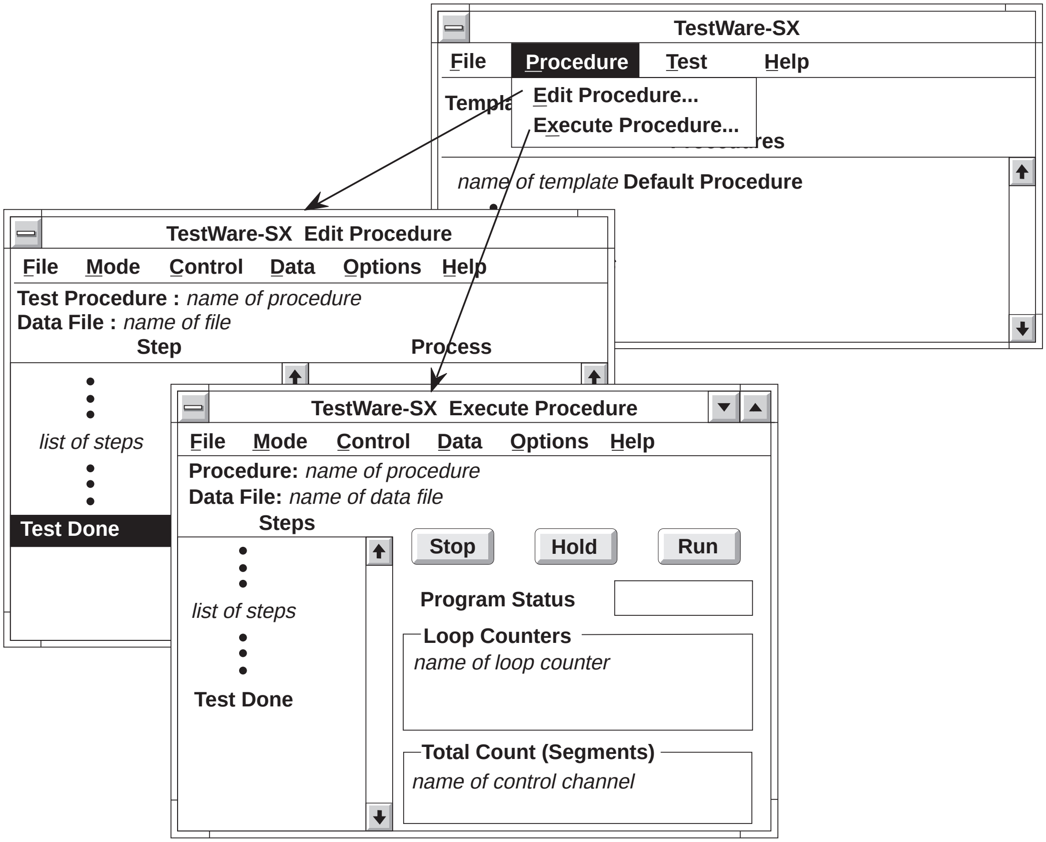


MENU	FUNCTION
Edit Procedure	Displays the Edit Procedure window where you define the parameters of the template for a specific procedure.
Execute Procedure	Displays the Execute Procedure window where you run a test.

Using the menu

Select a procedure, then select Edit or Execute in the Procedure menu. The appropriate window opens with the selected procedure as active.

Double-clicking a procedure in the TestWare-SX window functions the same as selecting Execute Procedure.

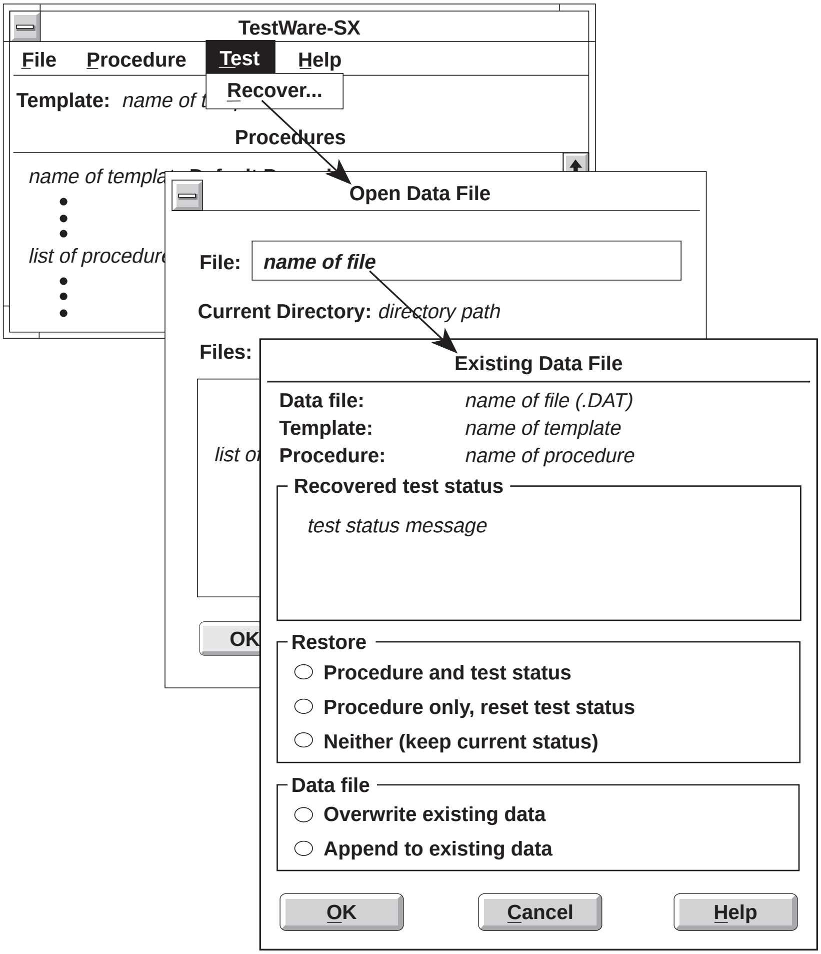


More information

Defining a procedure	*Edit Procedure Window* on page 199
Running a test	*Section B: Running a Test Procedure* on page 143 *Execute Procedure Window* on page 201

TestWare-SX Test Menu

Use the Test menu to access the test recovery feature so you can restore a test procedure that was stopped before it was completed.



Template Windows

More information

Auto save options	*Recovery Options Window* on page 193
Overview	*Section D: Test Recovery* on page 78
Restoration options	*Existing Data File Window* on page 168
Restoring a test procedure	*Section B: Running a Test Procedure* on page 143

Existing Data File Window

You can access this window from the Recover selection in the Test menu (main window) or the Open data file in the Data menu (Edit/Execute Procedure window).

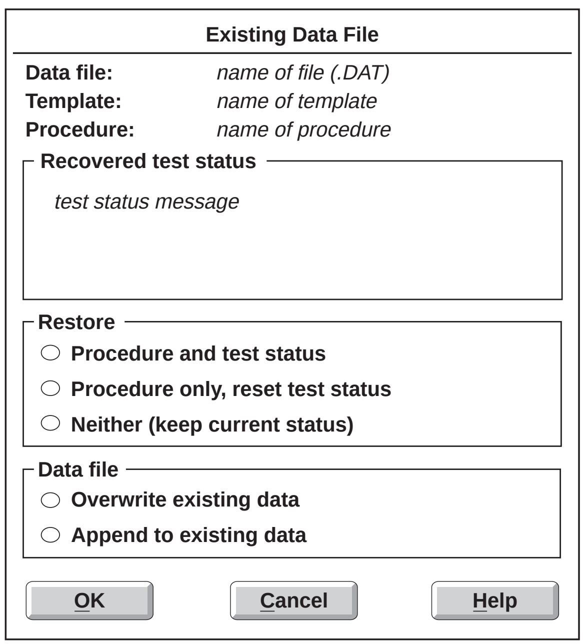
A test recover file has the same path and file name as the data file except it has the extension.SXS. There are three ways to use this window:

- Recovering from a test that stopped.
- Restoring a procedure that was modified during a test.
- Running a test (normal operation).

Prerequisite

You must open an existing data file that has a test recovery file associated with it to be able to use this window.

Use this window to select how the test procedure is restored and how the data file will acquire new data.



Control	Function
Data file: **Template:** **Procedure:**	These identify the template and procedure that are associated with the data file you opened.
Recovered test status	This area describes the status of the test when the last auto save function was performed. If no test status is found, restoration options are not available.
Restore	Restore the procedure and test status when you want to restore a test from the point where it was stopped. Restore the procedure only if you want to run the procedure again and overwrite the test status or, if you edited a procedure when you were executing it and did not save the revised procedure. Keep the current status for normal operation.
Data file	Use the overwrite option to reuse the data file when old data is no longer needed. Appending data to the end of the file is useful for recovering from a stopped test or iterations of a single test.

Recovering a test

You can recover from a test that stopped while it was running. The ability to recover from stopped test can vary.

- A test that was stopped intentionally should be easily recovered.
- A test was stopped by a catastrophic event such as a power failure or computer failure may have difficulty being recovered.
- The message in the Recovered test status area indicates if a test can be recovered.
- The accuracy of recovery is determined by the configuration of the auto save feature.

To recover a test, select **Procedure and test status**. Also, you will probably want to retain the existing data and append new data to the existing file. Press the OK pushbutton and the test will be restored at the point where the last auto save occurred. The program status will be stopped.

Template Windows

Restoring a procedure only

You can start a test, stop it, edit the procedure, then continue running the test (without saving the revised procedure). You may want to recover only the procedure but not the test you were running.

Note *For normal operation you can also use the Procedure only selection. You could use the Test/Recovery menu as a short cut to jump to the Execute procedure window with the procedure loaded and ready to run. You may need to reset the test from the Control menu if the test status is not reset.*

To recover the procedure, select **Procedure only**. This loads the procedure associated with the data file and creates a new test status file.

- If you want to execute the procedure, determine how you want to handle the data and select the appropriate choice. Then press the OK pushbutton and the test can start at the beginning of the procedure.
- If you want to save the procedure, select any data file option and press the OK pushbutton. Use the File menu and select Save As or Save As Template to save the procedure in the data base of templates and procedures.

Normal operation

Normal operation is when you want to run a test from a procedure you selected (as described in Chapter 3). If the procedure has been run previously, this window appears when you open the data file.

To run the selected procedure, select **Neither (keep current status)**. This uses the current procedure or one you intend to open and creates a new test status file.

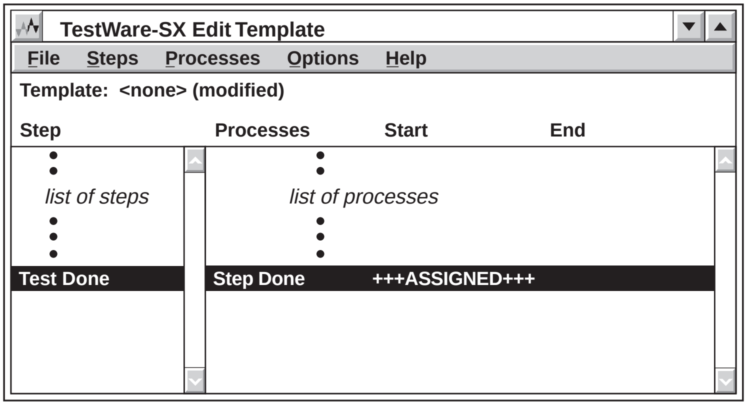
- If you no longer need the data currently in the data file, select Overwrite existing data and press the OK pushbutton.
- If you are running an iteration of tests, select Append to existing data and press the OK pushbutton.

More information

Auto save configuration	*Recovery Options Window* on page 193
How it works	*Section D: Test Recovery* on page 78

Edit Template Window

Use this window to create and edit test templates.



CONTROL	FUNCTION
File Menu	Lets you to create, open, save, and print templates.
Steps Menu	Lists the selections to edit steps in the template.
Processes Menu	Lists the selections to edit processes
Options Menu	Configures the data file format and auto save feature.
Template	Displays the name of the template you are working on. If you are creating a new template and have not saved it, the template name is <none>. When you change a template, [modified] follows the template name.
Step	Lists the step names of the template. Use the Steps menu to edit this list. Highlight a step to view its processes or edit it.
Process	Lists the processes associated with the selected step. Highlight a process to edit it. Each process lists what starts it and what stops it.

⚠ CAUTION

When you edit a template and save it, all associated test procedures are deleted. You will be warned before any procedures are deleted.

Use “Save as” to save the template with a different name.

Editing a Template

Use this window to create and edit test templates. When you create a new test template you add steps and processes in a sequence to accomplish your test requirements.

When you edit a test template you can change the sequence of steps and processes. You can also change the process parameters of the default procedure.

Note *The back side of the Designing a Test tab (Chapter 2) includes flow charts to create or edit a template. Chapter 2 also includes example procedures.*

Use the following as a guideline to establish a test template:

1. Use the File menu to create a new test template or open an existing test template.
2. Use the Steps menu to add new steps or edit existing steps in a test template. Refer to the Steps menu for information about editing steps.
3. Use the Processes menu to add new processes or edit existing processes in a step. Refer to the Processes menu for information about editing processes.
4. When you complete editing the test template, use the File menu and select Save to replace the existing file (deleting all associated processes) or select Save As to save the file with a different name or create a new test template file and retain the original file.

More information

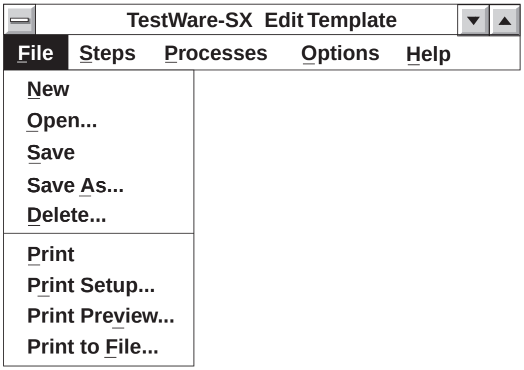
Creating and editing processes *Edit Template Processes Menu* on page 187

Creating and editing steps *Edit Template Steps Menu* on page 181

Opening and saving templates *Edit Template File Menu* on page 173

Edit Template File Menu

Use the File menu to open, create, or edit a template.



CONTROL	FUNCTION
New	Opens the Edit Template window without any steps or processes.
Open	Displays the Open Test Template window where you can select a previously saved template.
Save	Saves the current template file. If you have not assigned a name to the template, the Save Test Template window is displayed so you can name the template file.
Save As	Displays the Save Test Template window so you can save the file with a different name without changing the original file.
Delete	Displays the Delete Test Template window where you can remove a template file.
Print	Prints the current template/procedure to a printer. The information that is printed includes a list of all steps, processes, and their parameters.
Print Setup	Displays the parameters window of the default printer.
Print Preview	Displays the Print Preview window where you can view a list of the steps, processes, and parameters of the procedure.
Print to File	Displays the Print to File window where you can save the information that can be printed as a text file.

Template Windows

Printing overview

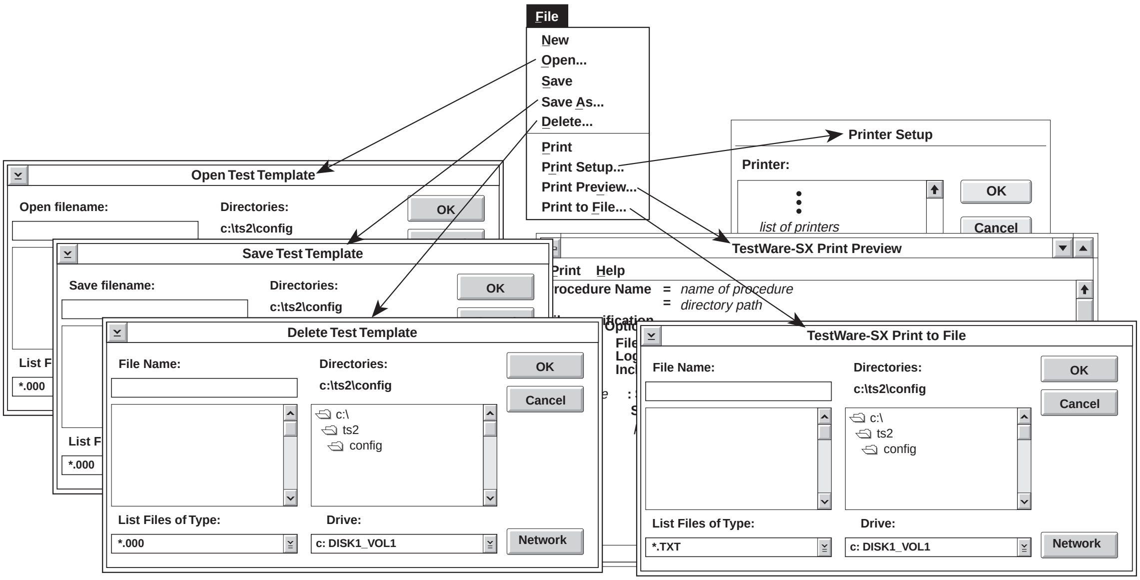
Printing a template allows you to obtain a printed copy of the selections and parameters that make up a test template. When you print a template, you are printing the default test procedure. Printed template information includes the following:

- The name of the template and its path.
- The data file options.
- The name of each step and the selected step done processes.
- The name of each process and all the parameters associated with the process design and parameter windows.

Printing a template

The following is a guideline to produce a printed copy of a template:

1. Open a template.
2. Select Print Setup in the file menu and select the printer to be used.
3. If you want to view the contents of the template, select Print Preview in the File menu.
4. If you want to maintain an electronic copy of the information that can be printed, select Print to File in the File menu.
5. Select Print in the File menu to obtain a printed copy of the template information.



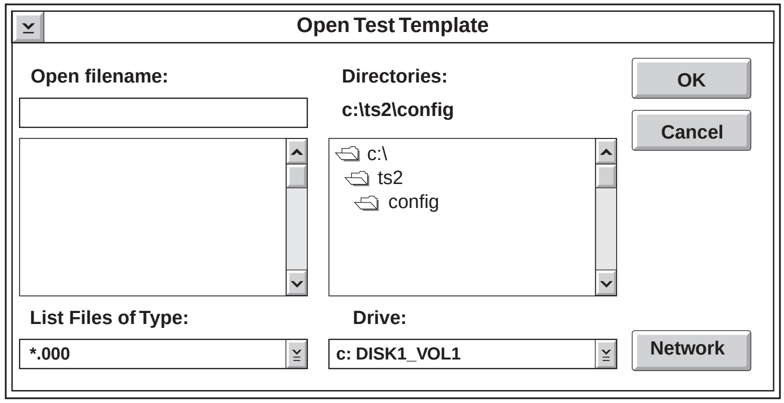
More information

Deleting a template	*Delete Test Template Window* on page 178
Opening a template	*Open Test Template Window* on page 176
Saving a template	*Save Test Template Window* on page 177
Saving template parameters as text	*Print to File Window* on page 180
Viewing all template parameters	*Print Preview Window* on page 179

Open Test Template Window

Use this window to open a template file.

Select the directory where your templates are saved, then select the template you want to open and press the Open pushbutton.

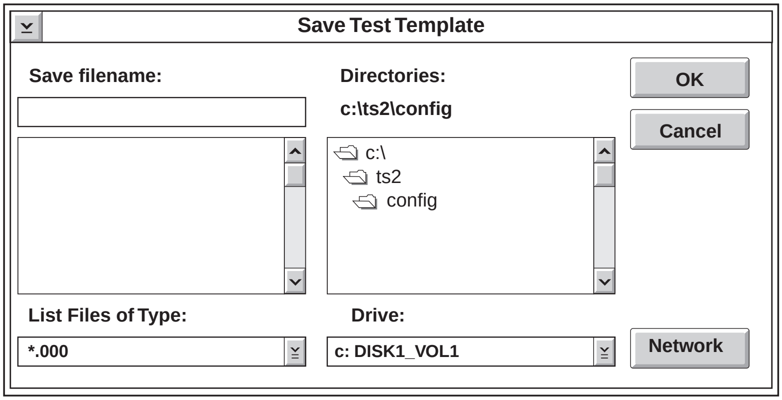


CONTROL	FUNCTION
Open filename	Displays ***.000** in the entry field. Select the file name of the template you want to open. All template files use the extension **.000**. When you select a file, its name is shown in the entry field
files	Lists the template files in the current directory. Selecting a file name displays it in the **File Name** entry field.
List Files of Type	Selects the type of files displayed in the File list. By default, ***.000** is selected. This displays only the files with the **.000** extension in the Files list.
Directories	Lists the available directories for the selected drive. Double-clicking a different directory displays the files of the directory in the **Files** list and any other directories in the **Directories** list.
Drive	Displays the current drive. All root directories of the drive are listed in the **Directories** list.
Network (Windows NT only)	Pressing the **Network** button displays the Connect Network Drive window where you can define new network drives and paths.

Save Test Template Window

Use this window to save a template file to disk.

Select the directory where you want to save the file, and type a name you want to call the template. Then press the Save pushbutton.



Control	Function
Save filename	Displays ***.000** in the entry field. Type the name you want to call the template. All template files use the extension **.000**. When you select a file, its name is shown in the entry field
files	Lists the template files in the current directory. Selecting a file name displays it in the **File Name** entry field.
List Files of Type	Selects the type of files displayed in the File list. By default, ***.000** is selected. This displays only the files with the **.000** extension in the Files list.
Directories	Lists the available directories for the selected drive. Double-clicking a different directory displays the files of the directory in the **Files** list and any other directories in the **Directories** list.
Drive	Displays the current drive. All root directories of the drive are listed in the **Directories** list.
Network (Windows NT only)	Pressing the **Network** button displays the Connect Network Drive window where you can define new network drives and paths.

Delete Test Template Window

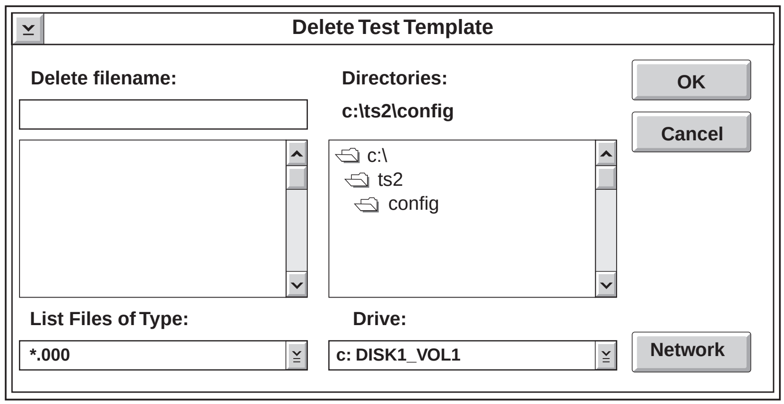
⚠ CAUTION

When you delete a template, all associated test procedures are also deleted.

Be certain that you want to delete the selected file.

Use this window to remove test templates you no longer need.

Select the name of the template you want to delete in the Files list. This displays the template name in the File entry field. Press the delete pushbutton to remove the template.

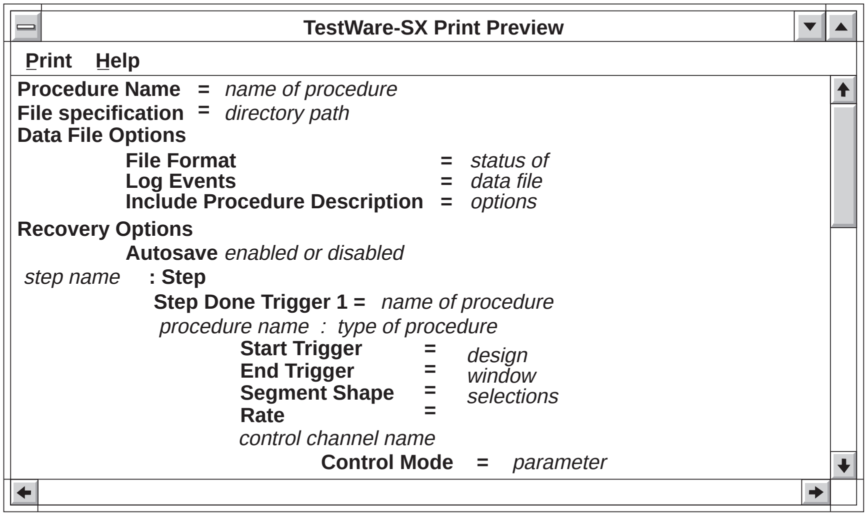


CONTROL	FUNCTION
File name	Displays ***.000** in the entry field. Select the file name of the template you want to delete. All template files use the extension **.000**. When you select a file, its name is shown in the entry field
files	Lists the template files in the current directory. Selecting a file name displays it in the **File Name** entry field.
List Files of Type	Selects the type of files displayed in the File list. By default, ***.000** is selected. This displays only the files with the **.000** extension in the Files list.
Directories	Lists the available directories for the selected drive. Double-clicking a different directory displays the files of the directory in the **Files** list and any other directories in the **Directories** list.
Drive	Displays the current drive. All root directories of the drive are listed in the **Directories** list.
Network (Windows NT only)	Pressing the **Network** button displays the Connect Network Drive window where you can define new network drives and paths.

Print Preview Window

The Print Preview window displays all the steps and processes that make up a template/procedure. The information includes all the parameters that define the template/procedure.

Use this window to view a summary of a template or procedure.



Using the window

The information displayed in the Print Preview window is an outline of the steps, processes, and their parameters that make up a template/procedure.

- Review the contents of the template/procedure for any errors.

  For example, check that each step has at least one process assigned to the step done trigger.

- Evaluate the start and end triggers for each process for proper sequencing.

  For example, be sure that segment command processes are sequenced in series. At least one process in a step must use the step start trigger.

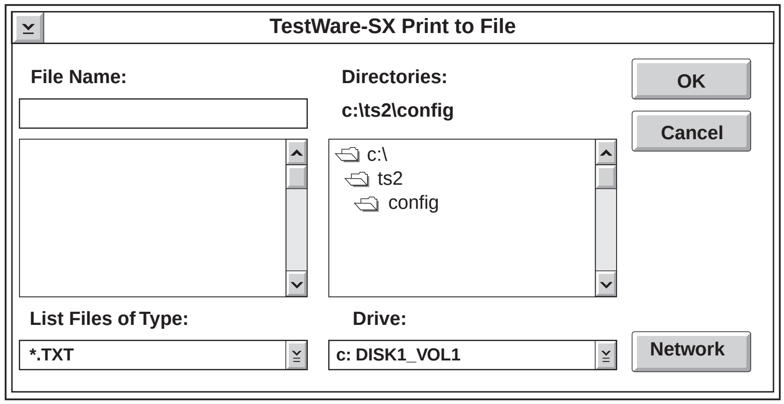
- The Print menu provides the same print functions as the File menu of the Edit Template and Edit Procedure windows.

Print to File Window

Test templates and test procedures are not text files. The Print to File window extracts information about the selected template/procedure and saves it in a text file.

Use this window to create a text file of the template/ procedure information.

Select the directory where you want the text file to be located. Press the OK pushbutton to save the template/procedure text file.

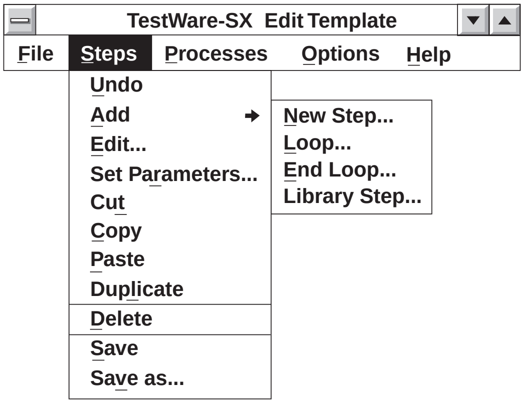


CONTROL	FUNCTION
File Name	Displays ***.TXT** in the entry field. Type the name you want to call the template file. All template files use the extension **.TXT**. When you select a file, its name is shown in the entry field
files	Lists the text files in the current directory. Selecting a file name displays it in the **File Name** entry field.
List Files of Type	Selects the type of files displayed in the File list. By default, ***.TXT** is selected. This displays only the files with the **.TXT** extension in the Files list.
Directories	Lists the available directories for the selected drive. Double-clicking a different directory displays the files of the directory in the **Files** list and any other directories in the **Directories** list.
Drive	Displays the current drive. All root directories of the drive are listed in the **Directories** list.
Network (Windows NT only)	Pressing the **Network** button displays the Connect Network Drive window where you can define new network drives and paths.

Edit Template Steps Menu

A step includes one or more processes. A step is a logical grouping of processes. Use steps to modularize a long test procedure into smaller logical units.

Use the Steps menu to create new steps or edit existing steps in a template.



SELECTION	FUNCTION
Undo	Allows you to cancel your last action in this menu.
Add	Displays an additional menu where you can select a new step, the loop function, or a library step.
New Step	Displays the Step Design window where you name the step.
Loop	Displays the Begin Loop Design window where you name the loop.
End Loop	Marks the end of the loop started with the Add Loop selection.
Library Step	This feature is not available.
Set Parameters	Displays the Loop Parameters window where you set the number of times the loop repeats.
Edit	Allows you to change the name of the step.
Cut	Removes the selected step and copies it into a buffer (replacing any previous buffer contents).
Copy	Copies the selected step into a buffer (replacing any previous buffer contents).
Paste	Puts a copy of the buffer contents at the location above the selection bar.

SELECTION	FUNCTION
Duplicate	Copies the selected step and pastes it below the selected step.
Delete	Removes the selected step.
Save	Saves the current step in the step library. Each time you save, the current file is overwritten.
Save as	Displays the Save Step window so you can save a step to the step library with a different name without changing the original file.

Undo

Use Undo to negate a menu action you have made. After you use Undo you cannot use it again until you make another menu action. The Undo menu selection also displays the menu action you can undo.

For example, you deleted a step. Select Undo Delete to restore the step. You could not select Undo again to delete the step again.

Adding a step

Adding a step inserts an empty step above the selection bar. The New Step selection displays the Step Design window where you name a new step. Use the Processes menu to define the processes in the step. A test template can include up to 50 steps and processes.

Adding a loop

Adding a loop inserts the name of the loop above the selection bar. Any steps entered after the loop name and before the End Loop designator are indented. These steps are repeated each time the loop repeats. Steps that are part of a loop are indented in the list of steps.

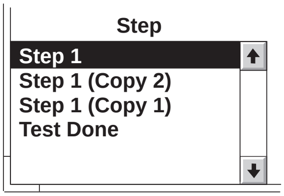
Selecting Loop displays the Loop Design window where you name the loop and loop counter. When you add a loop, you must select End Loop to complete the loop.

Editing a step

Using Cut and Paste allows you to move a step within the template or to a different template.

Using Copy and Paste allows you to move a copy of a step within the template or to a different template.

Duplicating a step functions like the combination of Copy and Paste. Duplicate adds (Copy #) to the name of the duplicated step.



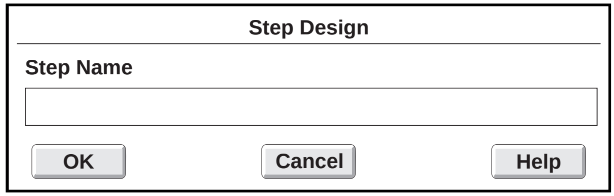
For example, duplicating a step called Step 1 results in a step called Step 1 (Copy 1). If you duplicate the step again, the new step is called Step 1 (Copy 2).

More information

Adding processes to a step	***Add a monotonic command*** on page 93 ***Edit Template Processes Menu*** on page 187
What are steps	***Terminology*** on page 53

Step Design Window

Use this window to name a step.



Using the window

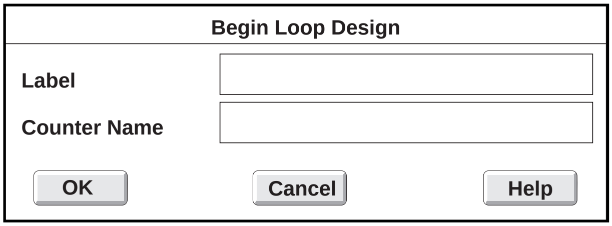
Type the name you want to call the step in the entry field. Press OK to assign the name to the step. Press Cancel to return to the Edit Template window without adding a step.

More information

What are steps *Terminology* on page 53

Begin Loop Design Window

Use this window to name a loop.



Using the window

Adding a loop or editing an existing loop (changing the loop name) displays the Step Design window. A loop represents a series of steps.

Type the name you want to call the loop in the Label entry field. The loop label is displayed in the Steps list of the template and procedure windows. Type the name you want to call the loop counter in the Counter Name entry field. The loop counter name is displayed in the Loop Counters list in the Execute window.

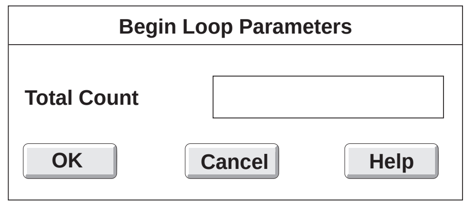
Press OK to assign the name to the step. Press Cancel to return to the Edit Template window without adding or changing a step.

Note *Remember, whenever you start a loop, you must end it with the End Loop menu selection.*

Begin Loop Parameters Window

Highlight the name of a loop and select Set Parameters or double-click the loop name to display this window.

Use this window to specify the number of times the loop is to repeat.



Using the window

Enter the number of loop repeats in the Total Count entry field. A count of 0 causes the loop to repeat continuously.

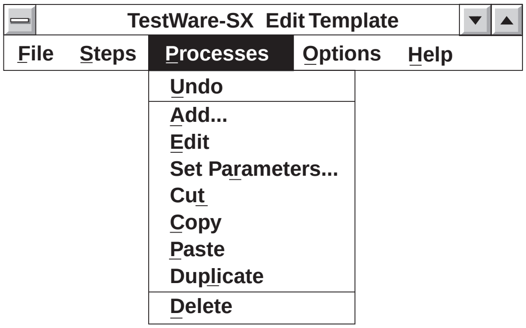
Press the OK pushbutton to enter the count. Press the Cancel pushbutton to return to the Edit Template window without entering a count.

Edit Template Processes Menu

Prerequisite

The template must have a step defined before you can use the Processes menu. A process must be inserted into a step.

Use the Processes menu to add or edit processes in a step.



Selection	Function
Undo	Allows you to cancel your last action in this menu.
Add	Displays the Select Process Type window where you select the process you want to add. The process is inserted above the selection bar.
Edit	Allows you to change the process information you entered with the Add menu selection (name, channel, triggers) for the selected process.
Set Parameters	Allows you to define the default procedure parameters of the selected process (control mode, segment shape, parameter values, etc.). This is the same as double-clicking the process name.
Cut	Removes the selected process and copies it into a buffer (replacing any previous buffer contents).
Copy	Copies the selected process into a buffer (replacing any previous buffer contents).
Paste	Puts a copy of the buffer contents at the location above the selection bar.
Duplicate	Copies the selected process and pastes it below the selected process.
Delete	Removes the selected process.

Undo

Select Undo to cancel the last menu action you have made. After you use Undo you cannot use it again until you make another menu action. The Undo menu selection also displays the menu action you can undo.

For example, you deleted a process. You select Undo Delete to restore the process. You could not select Undo again to delete the process again.

Adding a process

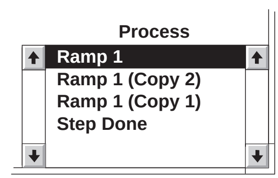
Before you add a process, be sure you have the desired step selected in the Steps list of the Edit Template window.

Editing a process

Using Edit displays the appropriate process window for the selected process.

Using Set Parameters displays the appropriate process parameters window for the selected process. This functions the same as double-clicking a process.

Using Cut and Paste allows you to move a process within the template or to a different template.



Using Copy and Paste allows you to move a copy of a process within the template or to a different template.

Duplicating a process functions like the combination of Copy and Paste. Duplicate adds (Copy #) to the name of the duplicated process.

For example, duplicating a process called Ramp 1 results in a process called Ramp 1 (Copy 1). If you duplicate the process again, the new process is called Ramp 1 (Copy 2).

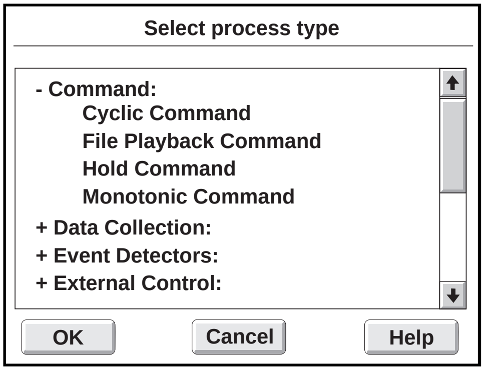
More information

Selecting a process	***Select Process Type Window*** on page 189

Select Process Type Window

Use this window to select a process and add it to a step.

Double-clicking a category name shows or hides the processes of the category.



Note *You may have additional processes listed in this window. Additional processes should have their own documentation.*

Using the window

Select the type of process you want to add to the template step and press the OK pushbutton. Pressing the Cancel pushbutton returns you to the Edit Template window.

Process types

TestWare-SX uses five types of processes. Each process is classified into one of the five categories.

- **Command** processes control the servovalve or motor using a closed loop system.
- **Data Collection** processes accumulate raw sensor data.
- **Event Detectors** processes either respond to detectors of conditions or create conditions that trigger other processes.
- **External Control** processes issue signals to devices external to the servo loop control system.
- **Special** processes can combine command, data collection, event detectors, and additional capabilities into one process.

Standard Processes

Category	Process	Function
Command processes	**Cyclic Command**	Repeats a waveform by assembling two monotonic segments and cycling them between two end levels.
	File Playback Command	Uses a data file to define a series of monotonic segments. Each segment contains information that defines a waveshape, a rate type and an end level.
	Hold Command	Holds the command for a specified time.
	Monotonic Command	Produces a segment command that starts at one level and ends at a different level.
Data Collection processes	**Data Acquisition**	A data acquisition process acquires one of three types of data. Peak/valley levels of each cycle Data at a specified time interval Data each time an input signal changes a specified amount
Event Detector processes	**Data Limit Detector**	Specifies a limit to trigger other processes.
	Digital Input Detector	Monitors up to 8 inputs from external sources to trigger other processes.
	Operator Event	Allows you to manually interact with the test procedure through a pushbutton.
External Control processes	**Analog Output**	Produces a voltage that can be output through one of the rear panel Readout connectors (J71 through J76).
	Digital Output Detector	Outputs up to eight separate 24-volt signals to external devices through the rear panel J55.
	Temperature Control	Communicates a temperature setting to an external temperature controller via an RS-232 interface.
Special processes	**Program Control**	Allows you to stop the test when this process executes. Use it like a custom interlock.

More information

About processes — *Section B: Processes* on page 54

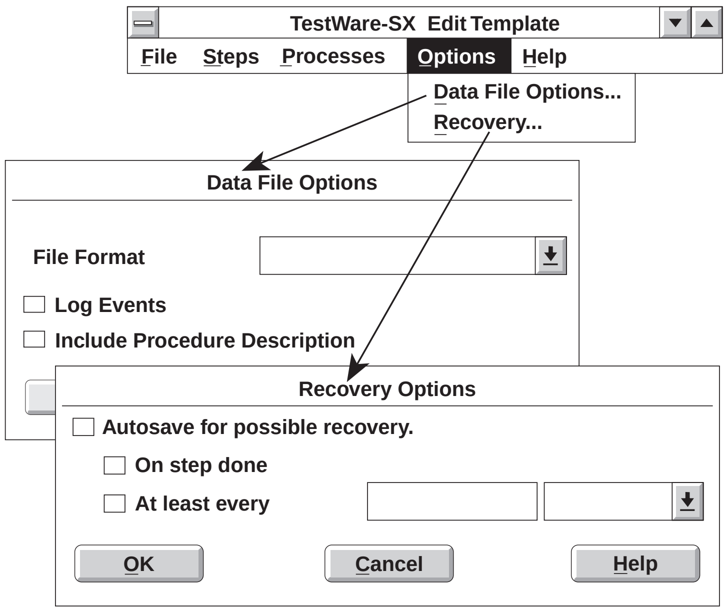
Process details — *The Processes* on page 227

Edit Template Options Menu

Use the Options menu to:

open the Data File Options window where you select file format

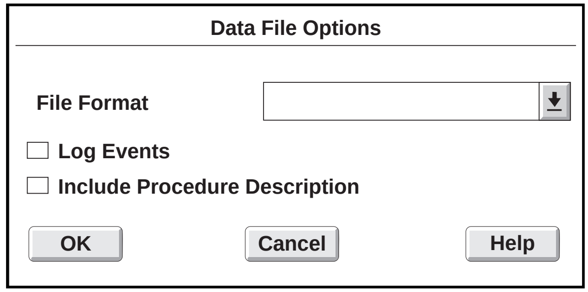
open the Recovery Options window where you specify how the auto save feature works.



Data File Options Window

Note *The default setting for the file format can be established when the TestWare-SX software is installed.*

Use this window to establish the file format for a data acquisition file.



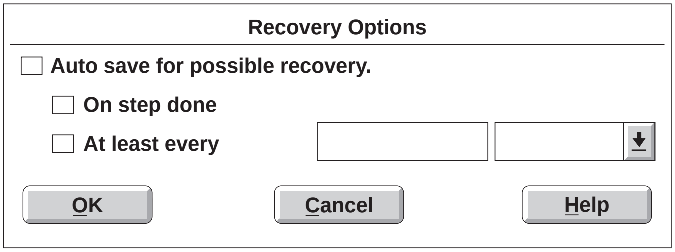
CONTROL	FUNCTION
File Format	Selects the file format to make the data acquisition file compatible with popular spreadsheet applications. Pressing the list icon shows the three file formats that are supported: ◆ Plain Text File ◆ Lotus Text File ◆ Excel Text File
Log Events	Records when you press the Run, Hold, and Stop pushbuttons. It also records interlocks that become active. The information is saved in the data file. Events are logged in the data file following the last data taken.
Include Procedure Description	Includes a listing of test procedure steps, processes, and parameters with the data file. The listing of the test procedure is the same information available with the Print Preview function in the File menu.

Using the window

Complete the Data File Options window before you open a data file. If the data file format is changed after the data file is opened, the original format will continue to be used. You will need to close the data file, then reopen it to use the new file format. The other attributes (check boxes) become active when the OK pushbutton is pressed.

Recovery Options Window

Use this window to configure the test recovery feature.



CONTROL	FUNCTION
Auto save for possible recovery	Enables the test recovery feature. The test recovery feature saves the test status any time the program status changes from run to stop or stop to run (including interlocks).
On step done	Saves the test status at the end of every step.
At least every	Defines how often a snapshot of the test status occurs. Type a value in the entry field and select the units with the list icon. Default is 15 seconds.

Using the window

When you create a new template, the Auto save feature is disabled. When you create a template you should decide if you want the test recovery feature enabled. Additional test procedures take on the attributes of the default procedure (the template). However, you can change the recovery settings for each procedure.

Note *Templates created under software versions 1.3 and older do not have default settings.*

For more information about test recovery, see Chapter 1, section D to find out how it works.

More information

How to recover a test — *Section C: Recovering a Test Procedure* on page 150

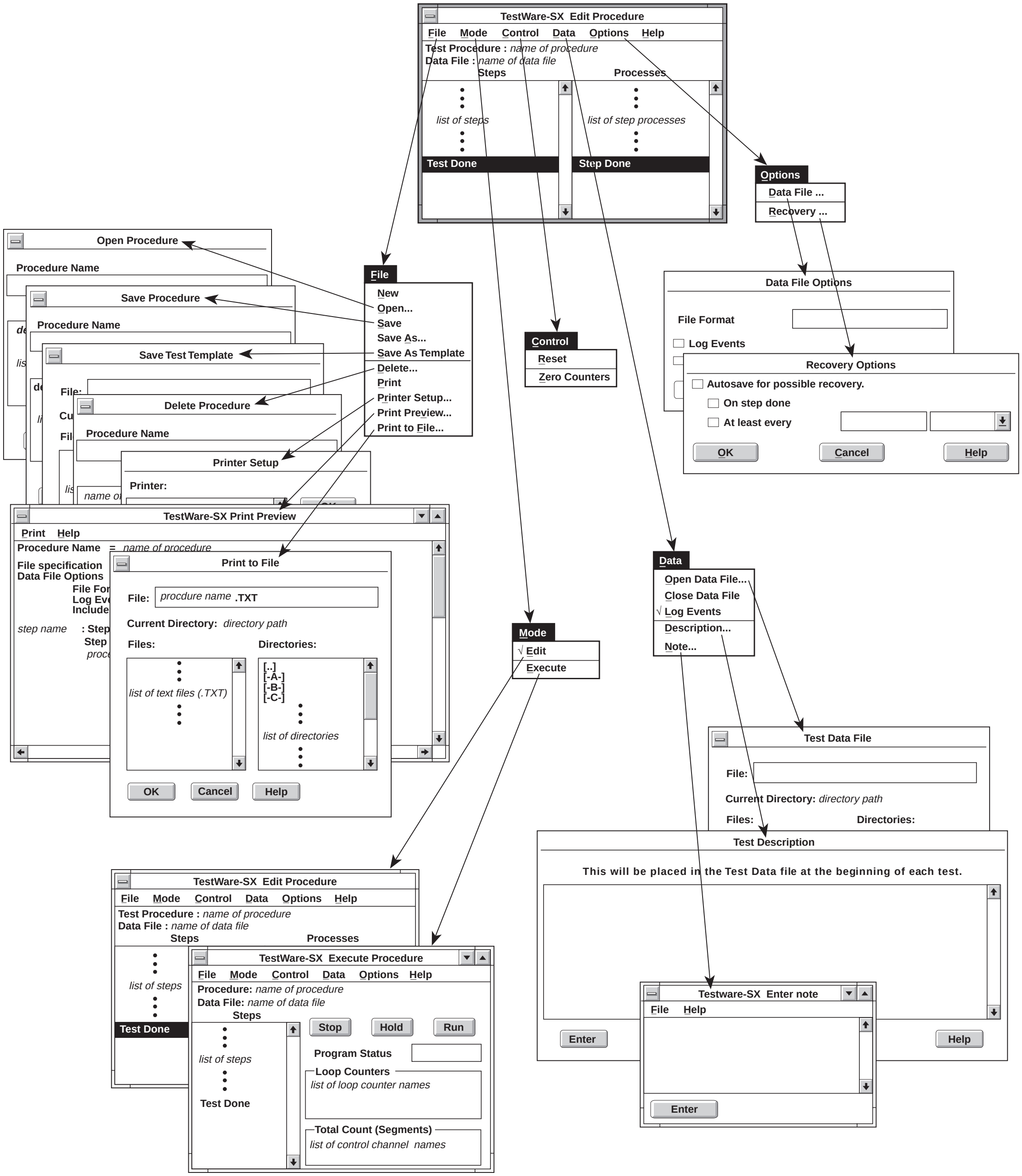
Test recovery overview — *Section D: Test Recovery* on page 78

MTS®

TestStar™ II

Control System

Chapter 5
Procedure Windows



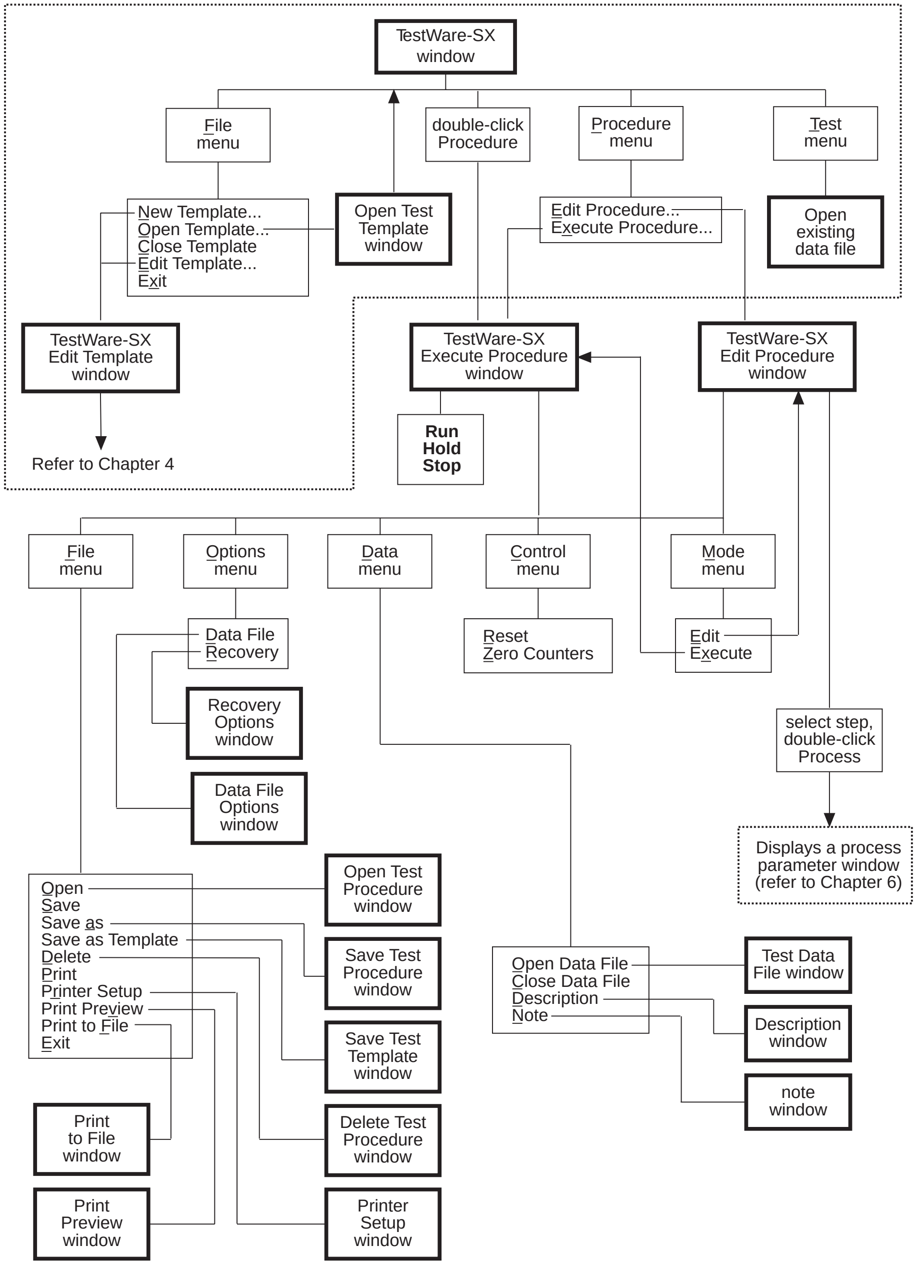
TestWare-SX Edit Procedure
File Mode Control Data Options Help
Test Procedure : name of procedure
Data File : name of data file
Steps
Processes
list of steps
list of step processes
Test Done
Step Done
Options
Data File ...
Recovery ...
Open Procedure
Procedure Name
Save Procedure
Save Test Template
Delete Procedure
Printer Setup
Printer:
File
New
Open...
Save
Save As...
Save As Template
Delete...
Print
Printer Setup...
Print Preview...
Print to File...
Control
Reset
Zero Counters
Data File Options
File Format
Log Events
Recovery Options
Autosave for possible recovery.
On step done
At least every
OK
Cancel
Help
TestWare-SX Print Preview
Print Help
Procedure Name = name of procedure
File specification
Data File Options
step name
Print to File
File: procdure name .TXT
Current Directory: directory path
Files:
Directories:
list of text files (.TXT)
[..]
[-A-]
[-B-]
[-C-]
list of directories
Data
Open Data File...
Close Data File
√ Log Events
Description...
Note...
Mode
√ Edit
Execute
Test Data File
File:
Current Directory: directory path
Files:
Directories:
Test Description
This will be placed in the Test Data file at the beginning of each test.
Enter
TestWare-SX Execute Procedure
Procedure: name of procedure
Data File: name of data file
Stop
Hold
Run
Program Status
Loop Counters
list of loop counter names
Total Count (Segments)
list of control channel names
Testware-SX Enter note
File Help

Chapter 5

Procedure Windows

This chapter includes the windows and menus related to editing and running procedures. Some descriptions include references to topics and their chapter numbers where more information can be found.

Contents



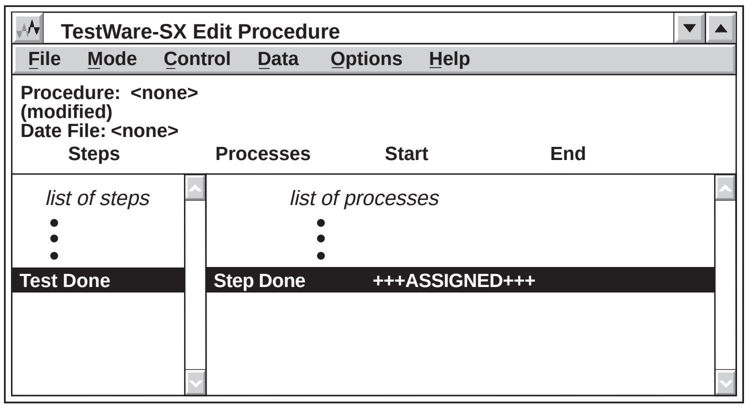
This diagram shows the relationships of the windows and menus of the edit procedure function of TestWare-SX. The darker boxes represent windows. The lighter boxes represent selections.

Edit Procedure Window

A test template can have several test procedures associated with it. The Edit Procedure window allows you to define the process parameters for a specific test.

Use this window to define process parameters.

Double-click the System Menu icon (upper left) to exit the procedure function and return to the main TestWare-SX window.



CONTROL	FUNCTION
menus	See the Procedure Menus section.
Procedure	Displays the name of the selected test procedure.
Steps	Displays the list of steps of the selected procedure. Highlight a step to view its processes.
Process	Displays the list of processes for the selected step. Double-clicking a process displays the appropriate process parameter window.

Editing a procedure

Use this window to create and edit test procedures. When you create a new test procedure you define each process of each step of the default test procedure. When you edit a test procedure you can change the parameters of one or more processes. Use the following as a guideline to establish a test procedure:

Note *The back side of the Designing a Test tab (Chapter 2) includes a flow chart to edit a procedure. Chapter 2 also includes a detailed edit procedure example.*

1. Use the File menu to create a new test procedure or open an existing test procedure.

2. Select a step (or add a step and add a process) then double-click a process. This displays the appropriate process parameter window for the process you selected.

3. Complete the process parameter window if you are creating a new test procedure. Change any parameters you choose if you opened an existing test procedure.

4. When you complete the test procedure definition, select "Save" in the File menu if you want to replace the existing file or select "Save as" if you want to save the file with a different name.

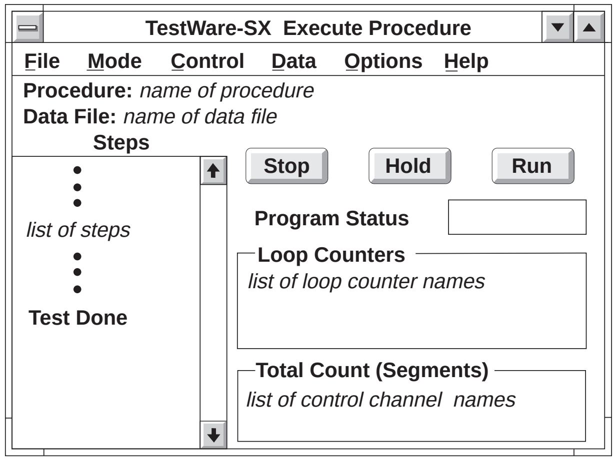
More information

Editing procedures	*Section B: Editing a Test Procedure* on page 115
Menu functions	*Procedure Menus* on page 204
Running a test	*Section B: Running a Test Procedure* on page 143 *Execute Procedure Window* on page 201
What are steps and processes	*Terminology* on page 53

Execute Procedure Window

Use this window to run a test procedure and monitor its progress.

Double-click the System Menu icon (upper left) to exit the procedure function and return to the main TestWare-SX window.



Control	Function
menus	See the Procedure Menus section.
Procedure and**Data File**	Displays the name of the selected test procedure and the data file.
Steps	Lists the steps of the selected test procedure. Each step is highlighted when it executes.
Stop	Pressing the Stop button ends the test.
Hold	Pressing the Hold button suspends the test until the Run or Stop button is pressed.
Run	Pressing the Run button starts the test.
Program Status	Displays the current test status. The status can be: ◆ Reset (Control menu) ◆ Running (Run button) ◆ Holding (Hold button) ◆ Stopped (Stop button) ◆ Done (test complete) ◆ Error (message)

CONTROL	FUNCTION
Loop Counters	Displays a list of the loop counters. Each loop counter shows the name of the loop and the number of times the loop has repeated.
Total Count (Segments)	Shows the number of segments that have been executed for each control channel. Reset the counters with the Control menu (Zero Counters).

Before running a test

You can select a different test procedure from this window using the File menu.

Before you press the Run button to start a test procedure, consider the following:

- You should have the specimen installed.
- You should have the TestStar configuration defined (sensors, detectors, display options, etc.).
- If your test includes data acquisition processes you need to set up a data file. Refer to the Data Menu descriptions.
- Use the Control menu to reset the test and counters.

During a test

While a test is running, the following are available to you:

- The step being executed is highlighted in the Step area of the window.
- You can press the Hold button to interrupt the test and hold the current output. Pressing the Run button resumes the test. You can also use the Test Control switches on the Load Unit Control panel to run, hold, and stop the test.
- You can stop the test at any time with the Stop button.
- If you opened a data file, you can type a note and enter it into the data file.
- You can monitor the number of segments that have been executed.
- You can adjust detectors and tune the servo control loop in the TestStar window.

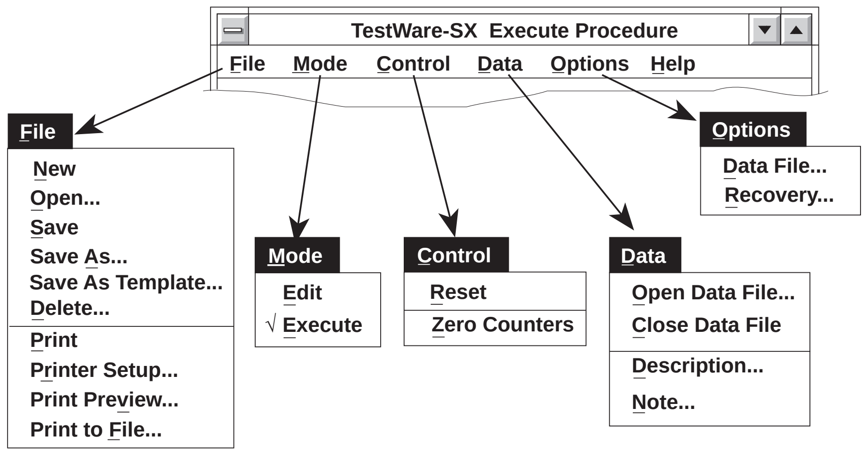
More information

Editing procedures	*Edit Procedure Window (5)* *Test Design (2)*
Menu functions	*Procedure Menus (5)*
Running a test	*Running a Test (3)*

Procedure Menus

These menus are the same for the Edit Procedure window and the Execute Procedure window.

The following sections describe each menu.



CONTROL	FUNCTION
File Menu	Allows you to create, save, open and delete test procedure files. You can also print or view the contents of a test procedure. This menu is disabled while the test is running.
Mode Menu	Allows you to change between the Execute Procedure and the Edit Procedure windows.
Control Menu	Resets the test and zeros the segment counter.
Data Menu	Creates or opens a data acquisition file and allows you to add test-related information to the data file. This menu is primarily used with the Execute Procedure window.
Options Menu	Establishes the format for the data acquisition file. The data file can be compatible with popular spreadsheet software. Configures the auto save feature. The auto save feature maintains a file with the test status (while the test is run). This file is linked to the data acquisition file and is used to restore a test.

More Information

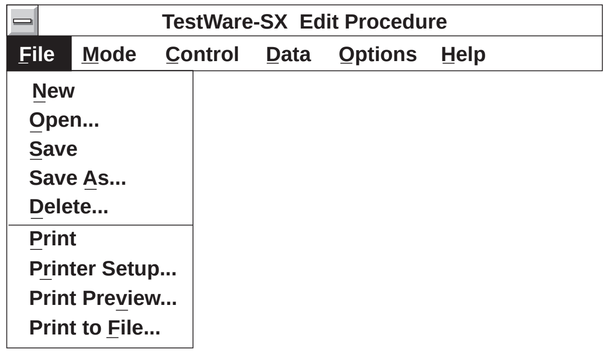
Restoring a test — *Section B: Running a Test Procedure* on page 143)

Test recovery overview — *Section D: Test Recovery* on page 78

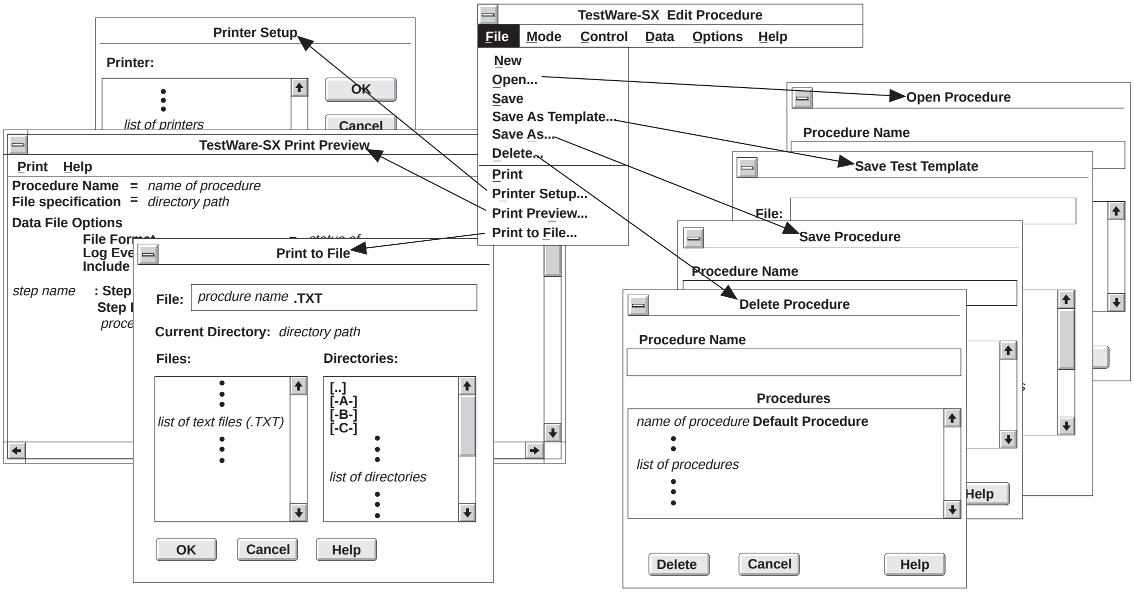
Procedure Windows

Procedure File Menu

Use the Edit Procedure File menu to create, save, open, and delete procedure files.



Menu Selection	Function
New	Allows you to create a new test procedure.
Open	Displays the Open Test Procedure window where you can select a previously saved test procedure.
Save	Saves the current procedure file. Each time you save, the current file is overwritten. If you have not assigned a name to the procedure, the Save Test Procedure window is displayed.
Save as	Displays the Save Test Procedure window so you can save the file with a different name without changing the original file.
Save as Template	Displays the Save Test Template window where the current procedure can be saved as a template.
Delete	Displays the Delete Test Procedure window where you can remove a procedure file you no longer need.
Print	Prints the current procedure. The information that is printed includes a list of all steps, processes, and their parameters.
Print Setup	Displays the Print Setup window where you select the printer you want to use.
Print Preview	Displays the Print Preview window where you can view the information that can be printed.
Print to File	Displays the Print to File window where you can save the information that can be printed as a text file.

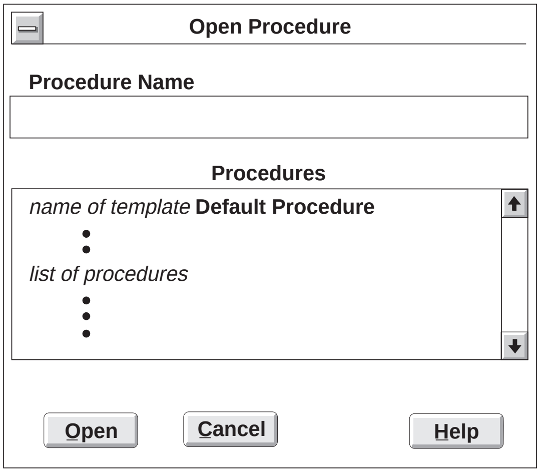


More information

Opening a file	*Open Procedure Window* on page 207
Saving a file	*Save Procedure Window* on page 208
Deleting a file	*Delete Procedure Window* on page 210)
Printing a file	*Print to File Window* on page 213

Open Procedure Window

Use this window to open a template procedure you want to edit or run.



Using the window

Select a procedure from the list of procedures. The procedures are listed alphabetically (not in the order you created them). The selected procedure is shown in the Procedure Name field.

Press the Open button (or double-click the selection) to open the procedure file. Press the Cancel button to return to the Edit Procedure window without opening a file.

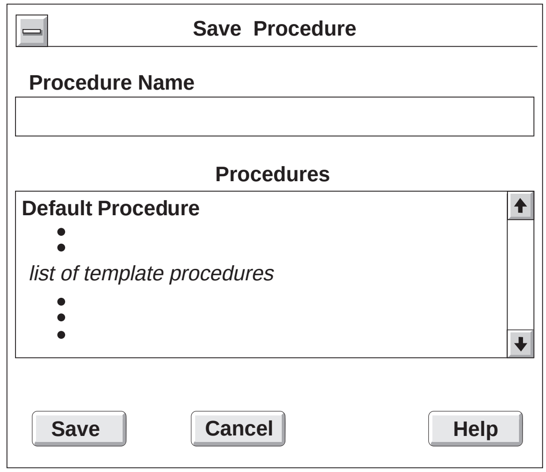
Procedure files

Procedure files have a numeric extension that is not displayed with this window. Since the template file (the default test procedure) uses the extension .000, each additional test procedure file associated with the template has a numeric extension that increases by one from the previously saved procedure.

For example, the third test procedure you make of a template named LCF would be LCF.003.

Save Procedure Window

Use this window to save a template procedure.



Using the window

Type a name you want to call the procedure in the Procedure Name entry field.

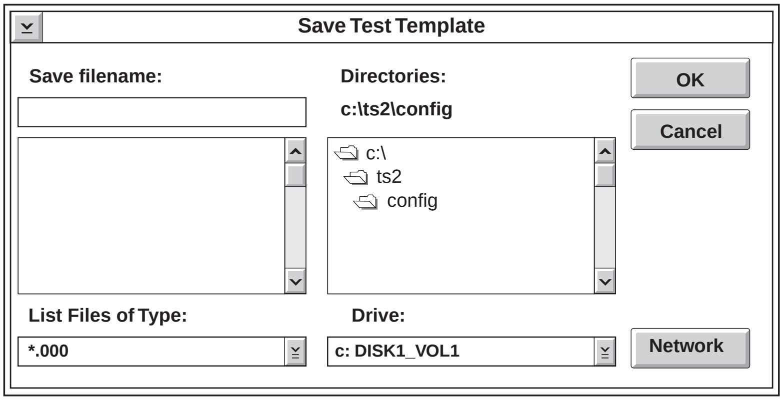
Press the Save button to save the procedure file. Press the Cancel button to return to the Edit Procedure window without saving the file.

Save Test Template Window

This is the same window used with the template function. The current procedure in the computer's memory is saved as a template file.

Use this window to save a procedure as a template file to disk.

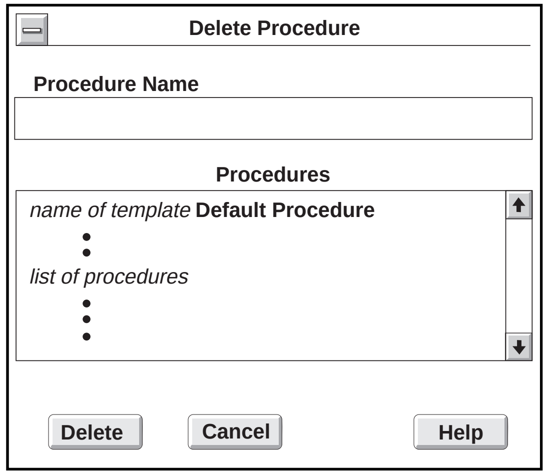
Select the directory where you want to save the file, and type a name you want to call the template. Then press the Save pushbutton.



Control	Function
Save filename	Displays ***.000** in the entry field. Type the name you want to call the template. All template files use the extension **.000**. When you select a file, its name is shown in the entry field
files	Lists the template files in the current directory. Selecting a file name displays it in the **File Name** entry field.
List Files of Type	Selects the type of files displayed in the File list. By default, ***.000** is selected. This displays only the files with the **.000** extension in the Files list.
Directories	Lists the available directories for the selected drive. Double-clicking a different directory displays the files of the directory in the **Files** list and any other directories in the **Directories** list.
Drive	Displays the current drive. All root directories of the drive are listed in the **Directories** list.
Network (Windows NT only)	Pressing the **Network** button displays the Connect Network Drive window where you can define new network drives and paths.

Delete Procedure Window

Use this window to remove test procedures you no longer need.



Using the window

Select a procedure from the list of procedures. The name of the procedure you want to delete is shown in the Procedure Name field.

Deleting the default test procedure displays a dialog box that informs you that the default test procedure cannot be deleted. You must use the delete template function to remove the default test procedure.

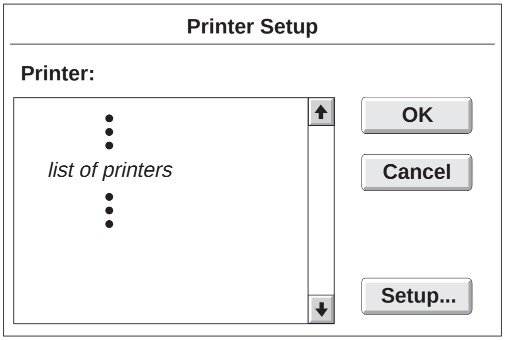
Press the Delete button (or double-click the procedure) to remove the procedure file. Press the Cancel button to return to the Edit Procedure window without deleting a file.

Printer Setup Window

Prerequisite

You must have print drivers installed for each available printer connected to your computer.

Use this window to select a printer.

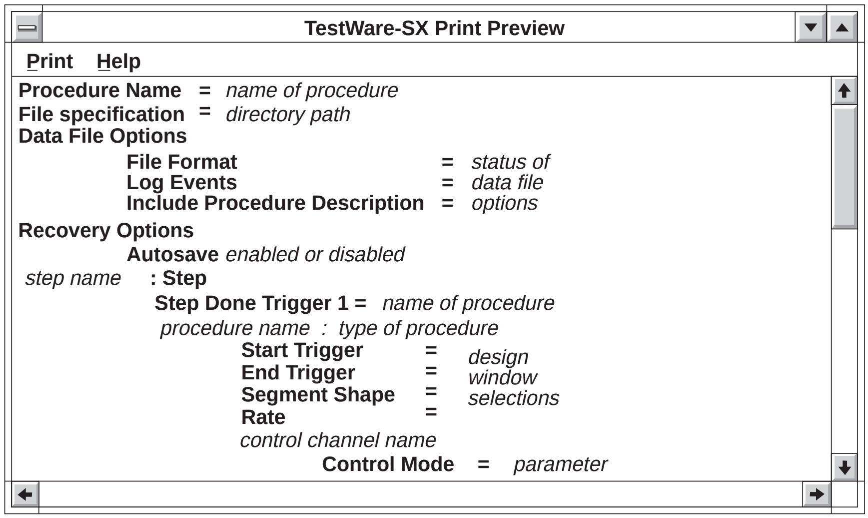


Using the window

If you have more than one printer, select the printer you want to use. Pressing the Setup button displays the Job Properties window. Use this window to establish the print options such as paper size, orientation, and scaling. Refer to your printer manual to determine the proper setup of the print manager.

Print Preview Window

The Print Preview window displays all the steps and processes that make up a template/procedure. The information includes all the parameters that define the template/procedure.



Using the window

The information displayed in the Print Preview window is an outline of the steps, processes, and their parameters that make up a template/procedure.

- Review the contents of the template/procedure for any errors.

  For example, check that each step has at least one process assigned to the step done trigger.

- Evaluate the start and end triggers for each process for proper sequencing.

  For example, be sure that segment command processes are sequenced in series. At least one process in a step must use the step start trigger.

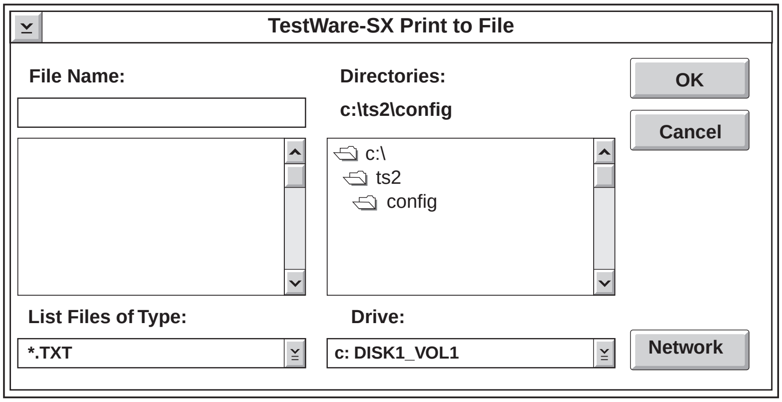
- The Print menu provides the same print functions as the File menu of the Edit Template and Edit Procedure windows.

Print to File Window

Test templates and test procedures are not text files. The Print to File window extracts information about the selected template/procedure and saves it in a text file.

Use this window to create a text file of the template/procedure information.

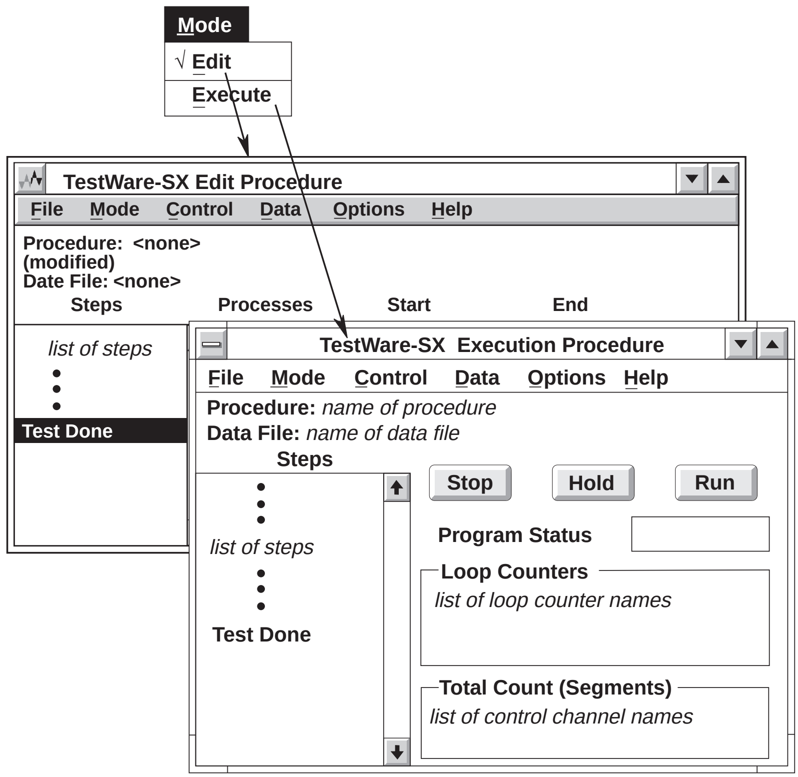
Select the directory where you want the text file to be located. Press the OK button to save the template/procedure text file.



CONTROL	FUNCTION
File Name	Displays ***.TXT** in the entry field. Type the name you want to call the template file. All template files use the extension **.TXT**. When you select a file, its name is shown in the entry field
files	Lists the text files in the current directory. Selecting a file name displays it in the **File Name** entry field.
List Files of Type	Selects the type of files displayed in the File list. By default, ***.TXT** is selected. This displays only the files with the **.TXT** extension in the Files list.
Directories	Lists the available directories for the selected drive. Double-clicking a different directory displays the files of the directory in the **Files** list and any other directories in the **Directories** list.
Drive	Displays the current drive. All root directories of the drive are listed in the **Directories** list.
Network (Windows NT only)	Pressing the **Network** button displays the Connect Network Drive window where you can define new network drives and paths.

Procedure Mode Menu

Use the Mode menu to move between the Edit Procedure and Execute Procedure functions.



Using the menu

- Select Edit in the Mode menu to display the Edit Procedure window where you can create, edit, and save procedures associated with a template.
- Select Execute in the Mode menu to display the Execute Procedure window where you can run a test procedure and acquire test data.

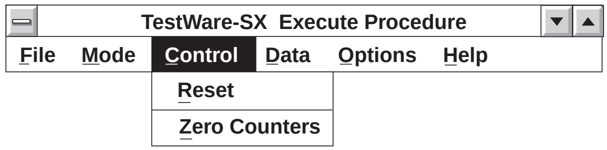
More information

Creating and editing procedure: *Execute Procedure Window* on page 201

Running a test procedure *Execute Procedure Window* on page 201

Procedure Control Menu

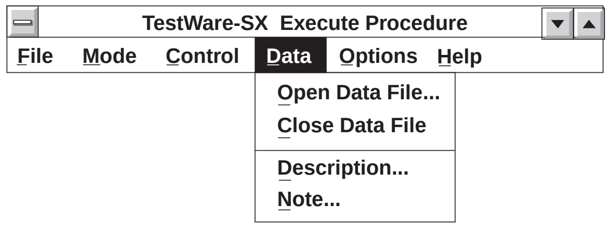
Use the Control menu to reset the test and zero the counters.



CONTROL	FUNCTION
Reset	Zeros all the counters (loops and segments) and resets the test. The test status shows Reset when this is used.
Zero Counters	Clears the segment counters in the Total Count (Segment) list without resetting the test.

Procedure Data Menu

Use the Data menu to establish a data acquisition file and record test related information.

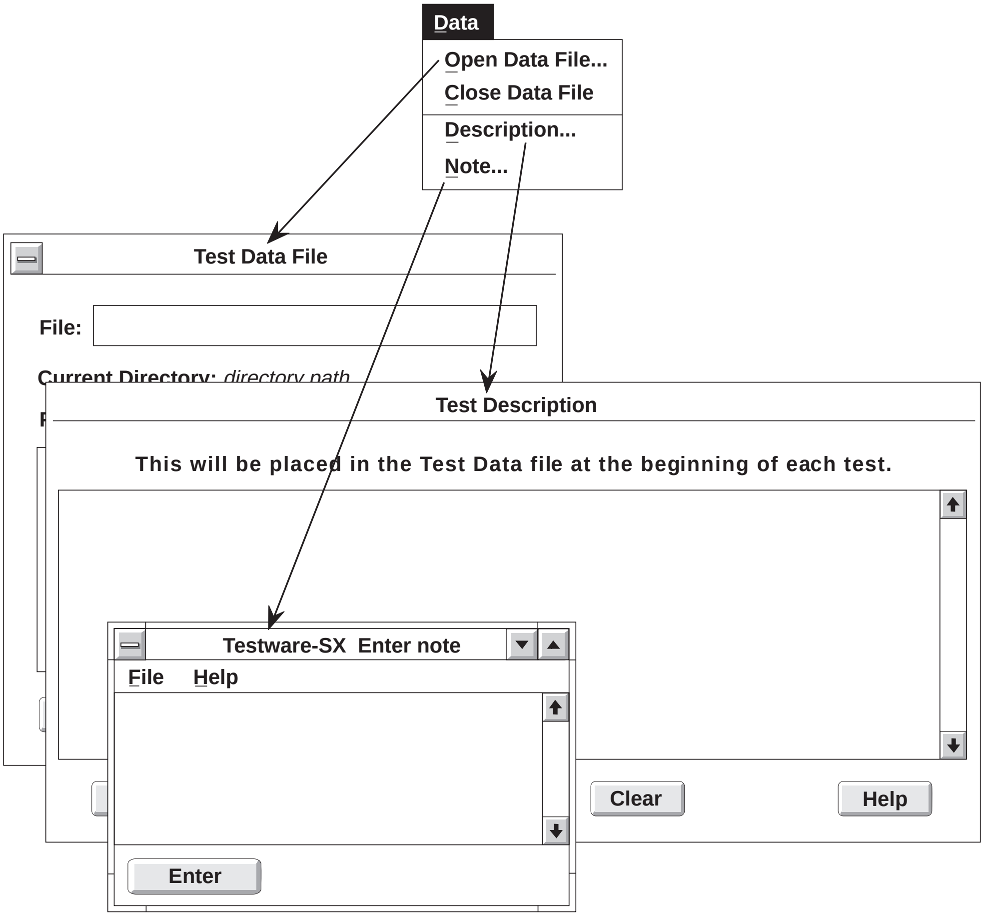


MENU SELECTION	FUNCTION
Open Data File	Opens an existing file or creates a new file to record data from data acquisition processes. You must open a data acquisition file before you can run a test that includes data acquisition processes.
Close Data File	Closes the file you opened for data acquisition. You close the file when the test is complete.
Description	Displays the Test Description window. This allows you to type a description of the test. The description is saved at the beginning of the data file.
Note	Displays a window where you can type information about the test progress. You use this window to record information about the test while it is running. Notes are saved in the data file following the last data taken.

Data File

A data file uses the extension .DAT. A second file is linked to the data file. This second file maintains the test status while it is being run. The linked file uses the extension .SXS.

When you open an existing data file another window opens that prompts you to select how you want the data written. You can overwrite the existing data or append new data to the existing data.



Procedure Windows

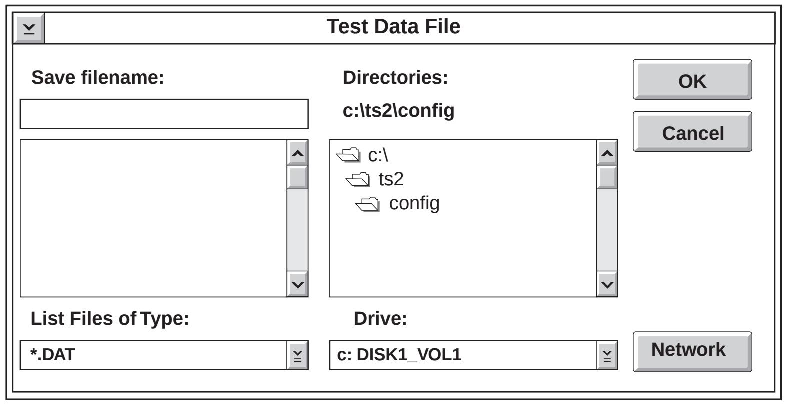
More information

Data file format	*Data File Options Window* on page 223
	Data Files on page 259
Opening a data file	*Test Data File Window* on page 218
Adding a test description to the data file	*Test Desciption Window* on page 220
Adding a note to the data file	*Operator Note Window* on page 221

Test Data File Window

You must open a data file if your test includes data acquisition processes. You should open a data file before you start your test. If you don't, you will be prompted to do so when you run the test (the test will not start until a data file is open).

.Use this window to add information about the test to the data file.



CONTROL	FUNCTION
File Name	Displays ***.DAT** in the entry field. Type the name you want to call the data file. All data files use the extension **.DAT**. When you select a file, its name is shown in the entry field
files	Lists the text files in the current directory. Selecting a file name displays it in the **File Name** entry field.
List Files of Type	Selects the type of files displayed in the File list. By default, ***.DAT** is selected. This displays only the files with the **.DAT** extension in the Files list.
Directories	Lists the available directories for the selected drive. Double-clicking a different directory displays the files of the directory in the **Files** list and any other directories in the **Directories** list.
Drive	Displays the current drive. All root directories of the drive are listed in the **Directories** list.
Network (Windows NT only)	Pressing the **Network** button displays the Connect Network Drive window where you can define new network drives and paths.

Opening a new file

1. Type the name you want to call the data file in the File entry field.
2. Select the directory where you want the data file located.
3. Press the OK button.
4. When the test is complete, return to the Data menu and select Close Data File.

Opening an existing file

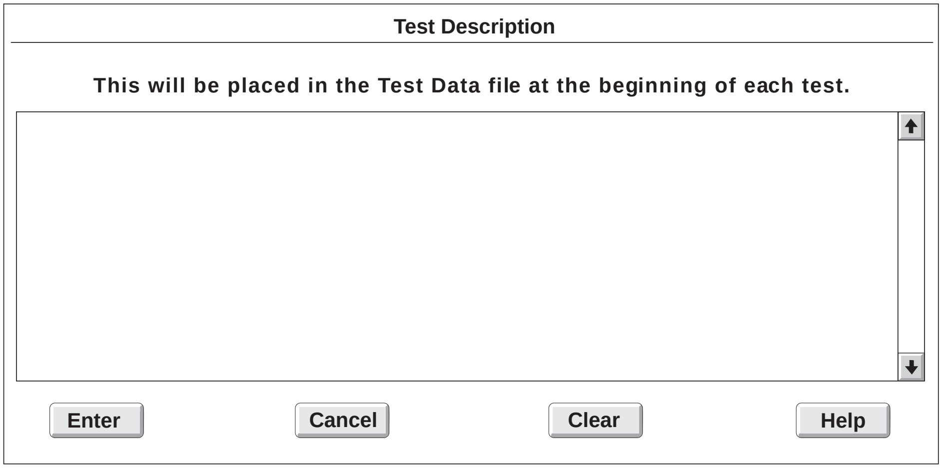
1. Select the directory where the data file is located. The default directory is C:\TS2\TWSX.
2. Select the data file from the list of data files. Only files with the extension .DAT are listed.
3. Press the OK button.
4. The Existing Data File window opens. Select Neither or Procedure only in the Restore area.
5. Select Overwrite or Append in the Data File area.
6. Press the OK button.
7. When the test is complete, return to the Data menu and select Close Data File.

More information

Data file format	*Data File Options Window* on page 223 *Data Files* on page 259

Test Description Window

Use this window to enter information about the test.



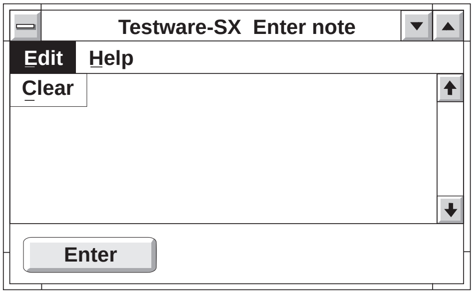
Using the window

Type the information about the test. The test description is added to the beginning of the data file.

- Press the Enter button to have the test description text included in the data file.
- Press the Cancel button to leave this window without changing a test description.
- Press the Clear button to clear the text from the window.

Operator Note Window

Use this to insert information about the test into the data file during a test.



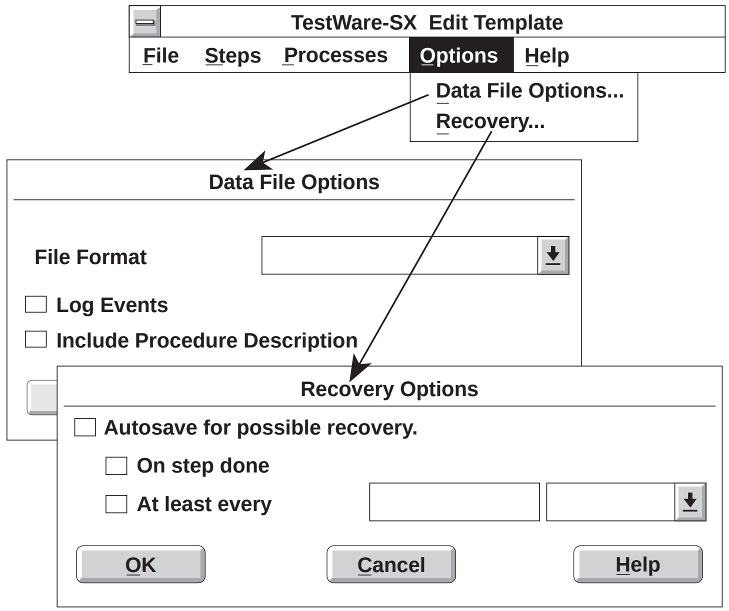
Using the window

Type the information you want inserted into the data file.

- Press the Enter button to have the note text inserted into the data file.
- Select Clear in the Edit menu to erase any existing note.
- Double-click the system menu icon (upper left corner) to close the window without changing the current note contents.

Procedure Options Menu

Use the Options menu to open the Data File Options window where you select file format or open the Recovery Options window where you specify how the auto save feature works.

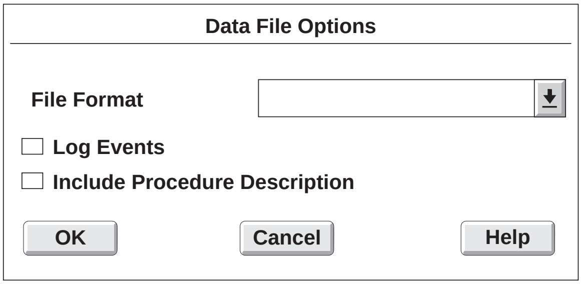


Procedure Windows

Data File Options Window

Note *The default setting for the file format can be established when the TestWare-SX software is installed.*

Use this window to establish the file format for a data acquisition file.



CONTROL	FUNCTION
File Format	Selects the file format to make the data acquisition file compatible with popular spreadsheet applications. Pressing the list icon shows the three file formats that are supported: ✦ Plain Text File ✦ Lotus Text File ✦ Excel Text File
Log Events	Records when you press the Run, Hold, and Stop buttons. It also records interlocks that become active. The information is saved in the data file. Events are logged in the data file following the last data taken.
Include Procedure Description	Includes a listing of test procedure steps, processes, and parameters with the data file. The listing of the test procedure is the same information available with the Print Preview function in the File menu.

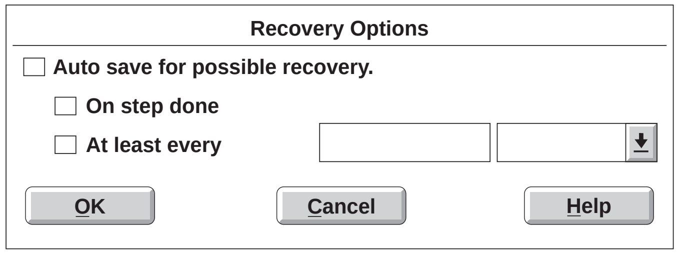
Using the window

Complete the Data File Options window before you open a data file. If the data file format is changed after the data file is opened, the original format will continue to be used. You will need to close the data file and then reopen it to use the new file format. The other attributes (check boxes) become active when the OK button is pressed.

Recovery Options Window

The Recovery Options window allows you to choose how to manage a test status file that can restore a test.

Use this window to configure the test recovery feature.



CONTROL	FUNCTION
Auto save for possible recovery	Saves the test status any time the program status changes from run to stop or stop to run (including interlocks). This also enables the options: • On step done • At least every
On step done	Saves the test status at the end of every step.
At least every	Saves the test status periodically. Defines how often a snap shot of the test status occurs. Type a value in the entry field and select the units with the list icon.

Default settings

When you create a new template, default settings are applied to the test recovery feature. Test procedures take on the attributes of the default procedure (the template). However, you can change the recovery settings for each procedure.

Note *Templates created under software versions 1.3 and older do not have default settings.*

More information

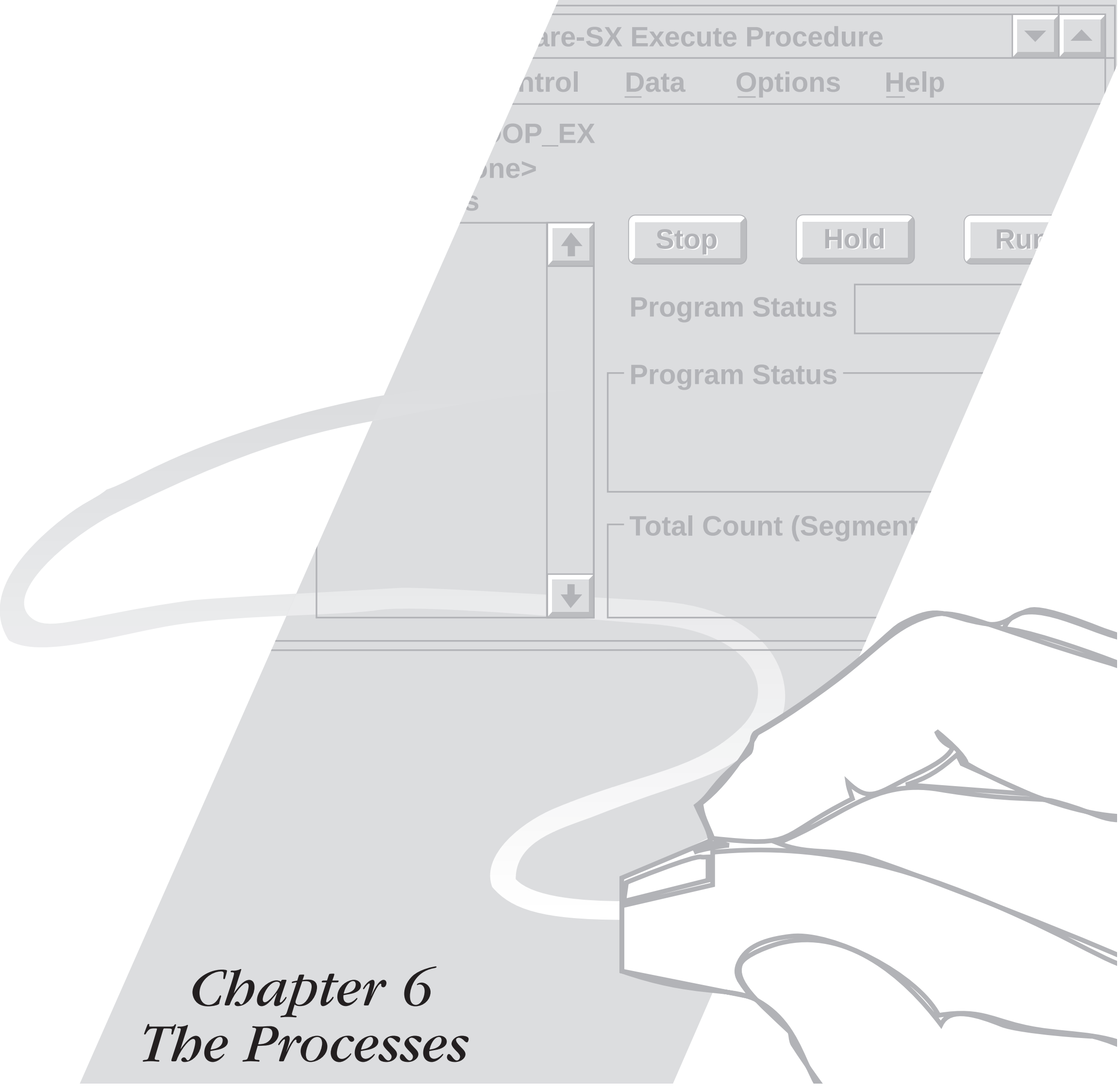
How to recover a test *Section C: Recovering a Test Procedure* on page 150

Test recovery overview *Section D: Test Recovery* on page 78

MTS®

TestStar™ II

Control System



Chapter 6
The Processes

Standard Processes

Command

Cyclic Command
External Command
File Playback Command
Hold Command
Monotonic Command

Data Collection

Data Acquisition
Temperature Data Acquisition

Event Detectors

Data Limit Detector
Digital Input Detector
Operator Event
Peak/Valley Change Detector

External Control

Analog Output
Digital Output
Temperature Control

Special

Program Control
Operator Information

Optional Processes

Command

Mixed Mode Pulse
Mixed Mode Sine
RPC File Playback
Run Time Ramp Control
UDA Cyclic

Data Collection

Dynamic Property Monitor
High Speed Data Acquisition
Run Time Plotting
Trend Monitor

Special

Dynamic Characterization
Resonant Search
Resonant Fatigue
Static Deflection
Tearing Energy

Chapter 6

The Processes

This chapter describes all the standard processes. You may have other processes that are part of optional TestWare applications.

Contents

Continued…

Contents (continued)

Continued…

Contents (continued)

Processes Overview

This chapter contains information about the standard set of processes. Optional processes are documented in other TestWare application manuals. This overview describes the following topics that all processes have in common:

- design and parameter windows
- default settings
- start and stop triggers
- how to reach the windows

Design and parameter windows

Each process uses at least two windows, a design window and a parameter window. Some processes include additional windows that are needed to define their function.

- Use the design window when creating a test template. It sequences the process with other processes. The design information is constant for all procedures associated with the test template. Design windows specify information associated with the test template. Each design window includes at least a name for the process along with the start and end triggers.

- Use the parameter window when creating test procedures. It defines values and selections for the unique set of parameters required by each process. The parameter information can be different for each test procedure. The parameter information is associated with a test procedure.

Default settings

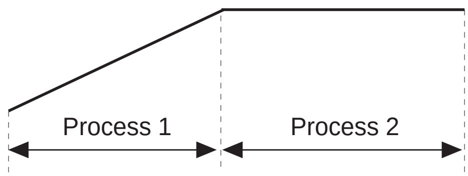
When you create a process for a template, default parameters and values are applied to the process. The default settings apply zeros to all value-entry fields and to the first selection from every list icon.

You have the option to apply the parameter settings to the default process in the Edit Template window. After you have created a process with the design window you can display the parameter window by double-clicking that process or selecting Set Parameters in the Processes menu. Assigning parameters while you are creating a template applies specific parameters to the default test procedure (the test template).

Using Triggers

All process design windows specify a start trigger and an end trigger to sequence the process within a step. You should draw a diagram of your test to figure out how to sequence the various triggers (refer to Chapter 2).

Triggers define how a process starts and how it ends.



	PROCESS 1	PROCESS 2
Start Trigger	Step Start	Process 1
End Trigger	none	none

For example, here Process 1 starts when the step begins (step start) and it ends when the process finishes its requirement which is ramp to a level (none). Process 2 starts when process 1 is done (start trigger process 1).

Other processes can be used as end triggers to stop a process before it completes its task.

Note *Sometimes it is helpful to add all the processes with default triggers, then edit them for the proper sequence.*

Default triggers

The default start trigger is *Step Start*. This trigger means the process begins when the step begins.

The default end trigger is <none>. This trigger means that the process continues until it does what it is supposed to do. A process with a none end trigger can be stopped prematurely if another process causes the Step Done process to complete the step.

Start triggers

The start trigger specifies when the process can begin.

- Use the Step Start trigger to start the process at the beginning of the step. All succeeding processes can be started by any process within the step.
- *For example*, assume the first command process is called Ramp Up. This process must use the start trigger “Step Start.”
- Use the name of any other process in the step to start the process when the selected trigger process ends.
- *For example*, assume you want to start a second process called Hold after the Ramp Up process is done. The start trigger for the Hold process would be “Ramp Up.”
- A command process must be sequenced in series with other command processes. Each command process should use the name of the preceding command process as the start trigger.

Note *Starting two test commands at the same time for the same control channel will display a message box that describes an error condition.*

- Except for command processes and some special processes, all other processes should be sequenced in parallel with command processes.
- Use the same start trigger as another process to run that process at the same time (in parallel).

End triggers

An end trigger stops the process.

- The default end trigger is <none>. The none trigger means the process will complete its task unless some other process causes the Step Done process to be encountered.
- You can use any process to stop another process before it completes its requirements. This causes the process to end when the trigger process ends.
- *For example*, assume you have a cyclic command process that cycles continuously and you want to stop cycling after some data is taken. The end trigger of the cyclic process would be the name of the data collection process.

Process Window Paths

Each process requires at least two windows: a design window and a parameter window.

- Design windows are used to create a template. These windows can only be reached when working with a template.
- Parameter windows are used to define a procedure. Parameter windows can be reached when either working with a template or with a procedure.

Template path

All the process windows can be reached when working with a template. Use the following path to reach any process window.

- Select the **File** menu in the **TestWare-SX** window,
- select **New Template**,
- select the **Steps** menu,
- select **Add** and **New Step,** then enter a step name,
- select the **Processes** menu and **Add**,
- select a process from the **Select Process Type** window and complete the design window for that process,
- double-click the name of the process you named with the design window to display the parameters window of the default procedure.

Procedure path

Use the following path to reach any parameter window when working with a procedure.

- Select the **File** menu in the **TestWare-SX** window,
- select **Open Template**,
- double-click a template (or select a template and press **OK**),
- select a procedure, then select the **Procedure** menu and **Edit Procedure**,
- select a step that includes the desired process,
- double-click the name of the desired process to display the parameters window.

Analog Output

The Analog Output process is an external control process that produces an analog voltage to a rear panel Readout connector. This process can be used as a remote set point command for an external device.

Note *Pages 230 through 233 provide details common to all processes such as default settings, using triggers, and how to reach the process windows.*

Contents

Prerequisites

You must configure an output signal for the Programmable function (see the Edit menu in the Reference manual).

You should have an external device connected to the rear panel connector configured for the programmable function (see the Cabling chapter in the Installation manual).

You need to know what voltage range can be input to the external device.

How it works

An analog output process can set an output voltage for a single output signal.

- When the process starts, the output steps to the specified voltage level and ends the process. This process executes very quickly.
- When the process ends, the voltage output remains at the specified level. Create another process to change the output voltage.
- The output can be scaled using the Channel Setup button.

About the external device

You need to know some information about the external device that will receive the analog output voltage. The external device should be connected to one of the rear panel Output connectors. These are BNC connectors. Determine the following:

- What type of units are appropriate for this device?
- *For example*, a temperature controller would use degrees (such as Celsius or Fahrenheit).

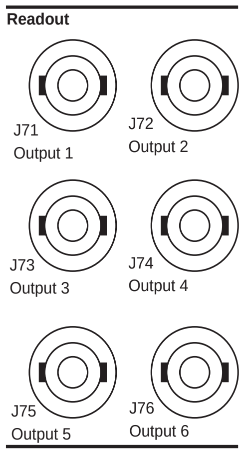
What are the minimum and maximum input voltage range of the device?

- *For example*, if the device accepts a 0 to 5 volt input you will need to determine how to scale the ±10-volt analog output to avoid damaging the device.
- The rear panel connectors are located on the rear panel of the digital controller.

  Connectors J71 through J76 correspond with output signals 1 through 6 respectively.

  You can monitor all six output signals at connectors J41A and J41B.

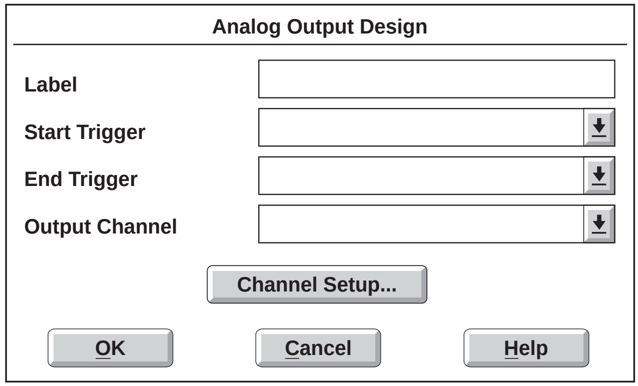
  Output signals must be configured for the Programmable function with the Edit Output window -- See the TestStar Reference Manual.



Analog Output Design Window

The information in this window is saved with the test template.

This window names the process and defines what starts and stops it.



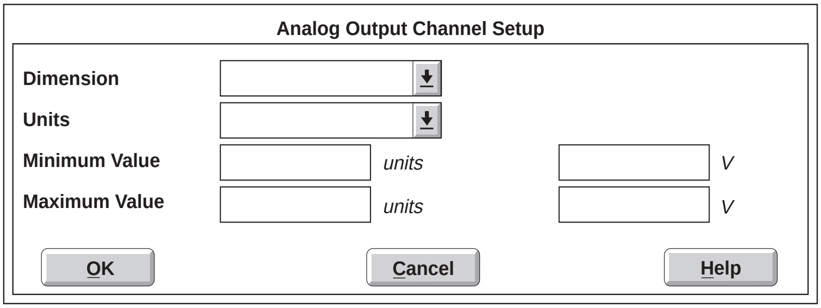
CONTROL	FUNCTION
Label	Names the process. Type the name you want to call the process in the entry field.
Start Trigger	Specifies the beginning of the process. Press the list icon and select a trigger.
End Trigger	This process occurs so quickly it is not practical to use another process to stop it. Use the default end trigger <none>.
Output Channel	Selects the output signal for the process. Pressing the list icon shows the output signals that are configured as programmable.
Channel Setup	Displays the *Analog Output Channel Setup Window* on page 237 where you scale the output voltage.

Analog Output Channel Setup Window

The default settings for this window set the dimension to volts with a range (min./max.) of ±10 volts. The information in this window is saved with the test template.

Note *You need to know the characteristics of the external device that you intend to use with the analog output before you can complete this window.*

Use this window to scale the voltage output and select the units for an analog output process.



CONTROL	FUNCTION
Dimension	Specifies the type of units for the output.
Units	Specifies the specific units for the output The units define the minimum and maximum values.
Minimum Value	Assigns the lowest output unit value to the lowest output voltage. The minimum voltage can be -10 volts.
Maximum Value	Assigns the highest output unit value to the highest output voltage. The maximum voltage can be +10 volts.

Using the window

The Channel Setup window assigns a dimension and defines an output range that is used with the Analog Output Parameters window (where you set the output value).

1. Select the type of dimension for the channel. The type of dimension determines the kinds of units available for the Analog Output values. Choose a dimension that is appropriate for the device receiving the output.
2. Select a unit for the minimum and maximum values.
3. Enter a minimum value and a minimum voltage. This defines the lowest setting available for the output.
4. Enter a maximum value and a maximum voltage. This defines the highest setting available for the output.

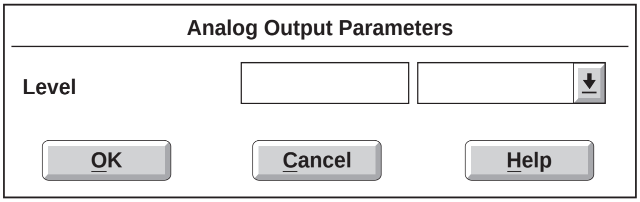
For example, suppose you want to output a remote set point to a temperature controller. Also assume that the device accepts a 1 to 10 volt signal representing 0 to 1100° C.

- Select temperature as the dimension and deg-C for the units.
- Set the minimum value at 0° C and 1 volt. Set the maximum value to 1100° C and 10 volts. This scales the output so 1 volt = 0° C and 10 volts = 1100° C.
- You can specify any value between 0° C and 1100° C or 1 volt and 10 volts with the Analog Output Parameters window.

Analog Output Parameters Window

You must establish an analog process with the Analog Output Design window and scale the output with the Channel Setup window before you can use this window. The information in this window is saved with the test procedure.

Use this window to set the voltage level of the analog output process.



Using the window

- Use the list icon to select the units for the output level.
- Type the value for the output level in the entry field.

Cyclic Command

A cyclic command process produces a command signal that controls a servovalve or servo motor. This process should be sequenced in series with other command processes.

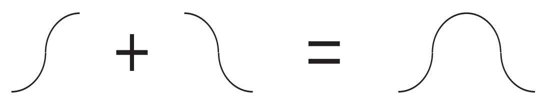
Note *Pages 230 through 233 provide details common to all processes such as default settings, using triggers, and how to reach the process windows.*

Contents

How it works

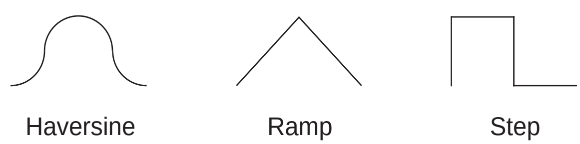
A cyclic command creates a waveform by assembling two single segments and repeating them continuously or for a predefined number of cycles.

One cycle works like two monotonic commands.



Three types of waveforms can be created.

A cyclic waveform can have one of three shapes.



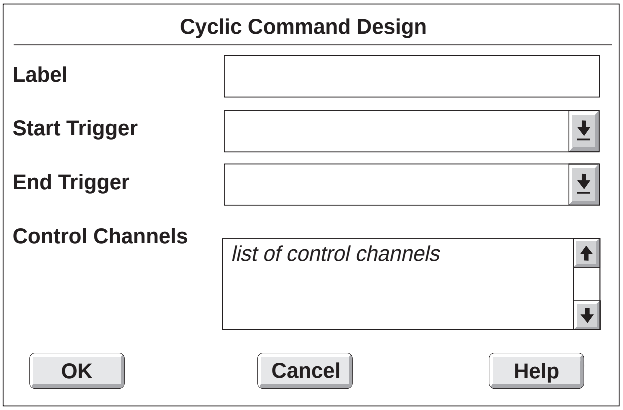
The Processes

Cyclic Command Design Window

This window names the process and specifies how the process starts and stops.

Note *Starting two test commands at the same time for the same control channel will display a message box that describes an error condition.*

The information in this window is saved with the test template.



CONTROL	FUNCTION
Label	Names the process. Type the name you want to call the process in the entry field.
Start Trigger	Specifies the beginning of the process. Press the list icon and select a trigger.
End Trigger	Specifies the end of the process. Press the list icon and select a trigger.
Control Channels	Specifies which control channels the process is to be applied (more than one may be selected). A list of the available control channels is shown.

Trigger example

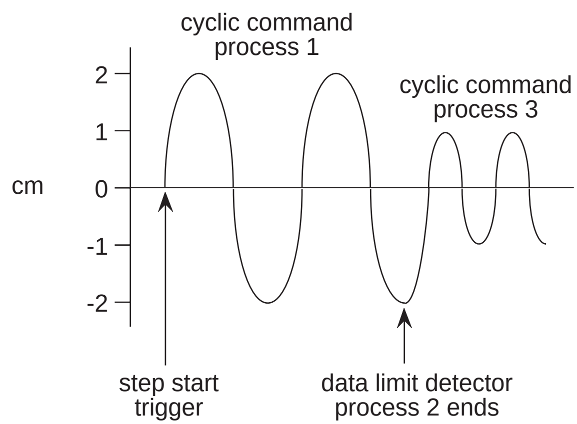
Assume you created three processes with the following attributes (only the attributes relevant to the example are listed):

Process 1	Process 2	Process 3
start trigger = step start	start trigger = step start	start trigger = process 1
end trigger = data limit	end trigger = none	end trigger = none
end level 1 = +2 cm	channel = segment count	end level 1 = +1 cm
end level 2 = -2 cm	limit value = 4 segments	end level 2 = -1 cm
repeats = 10 cycles		repeats = 2 cycles

Cycle process 1 begins at the same time the data limit detector process 2 starts.

When process 2 detects 4 *segments* it ends. When process 2 ends, process 1 also ends, even though only 2 cycles were completed.

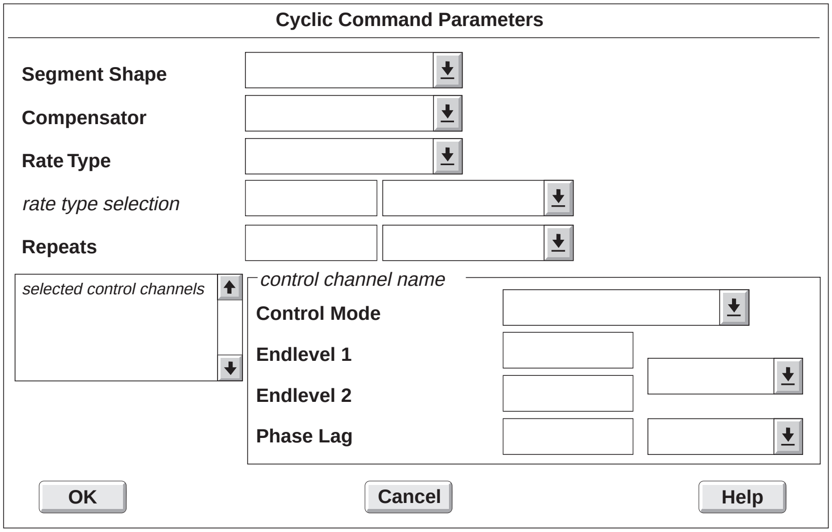
The end of process 1 starts cyclic process 3.

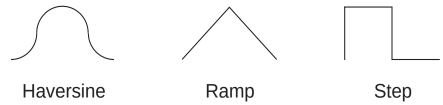


Cyclic Command Parameters Window

You must establish a cyclic process with the Cyclic Command Design window before you can use this window. The information in this window is saved with the test procedure.

Use this window to define the specific characteristics of the cyclic waveform.



Control	**Function**
Segment Shape	Lists the three segment shapes you can select. The relative segment shapes let you define the command amplitude with relative end levels. 
Amplitude/Mean Control	Enables the amplitude and mean control function. This function ensures that programmed amplitudes and mean levels are achieved.
Rate Type	Lists the types of expressions you can select to define a cycle. The rate type allows you to specify the units you prefer to define a cyclic waveform. The types of expressions are: • Frequency • Time • Rate
rate type selection **•Frequency** **•Time** **•Rate**	Specifies the cyclic command value. The label for this field depends on the selected Rate Type. Pressing the list icon of the right entry field shows the units you can select for the rate type value. The type of units depends on the selected Rate Type. Type the value you want to specify for the cyclic command in the left entry field.
Repeats	You can express repeats with segments or cycles. Pressing the list icon of the right entry field shows the segment and cycle selections. Select segment or cycle and type the number of repeats you want.
selected control channels	Displays a list of the control channels selected in the Design window.
Control Mode	Selects the control mode for the cyclic command. Pressing the list icon shows the control modes defined with the Controller Definition program.
End level 1 **End level 2**	Specifies the amplitude of the cycle (the difference between End level 1 and End level 2). The cyclic command goes to End level 1 first. End levels can be specified with absolute values or relative values (depending on the segment shape selection). Pressing the list icon of the right entry field shows the units you can select for the end levels.
Phase Lag	Specifies how much the cyclic command of the selected control channel leads or lags another control channel. Press the list icon of the right entry field to select the phase units (degrees or radians). Type the amount of phase into the left entry field. A positive value lags the reference control channel and a negative value leads the reference control channel.

Compensators

Compensators compare the sensor feedback to the test command to determine if the command signal is actually being properly applied to the specimen. Compensators can provide specific types of correction to ensure the command is properly applied to the specimen.

Note *Be sure to disable any compensation when tuning.*

The convergence rate determines how fast the compensators work. See the Adjust Compensators window in the TestStar Reference manual for related adjustments.

Peak/Valley

This compensator detects any amplitude roll-off and any difference in the mean level. Amplitude roll-off refers to the tendency of amplitudes measured by the sensors to be less than the desired amplitudes. This can also cause a mean level difference.

Enabling the peak/valley compensation causes the program to adjust the servovalve command signal until the programmed amplitude is achieved and the mean level is maintained. It takes a few cycles to start amplitude/mean control and about five cycles to end it.

PAC

This compensator detects any amplitude roll-off and any phase lag in sinusoidal waveforms. Amplitude roll-off refers to the tendency of amplitudes measured by the sensors to be less than the desired amplitudes. Phase lag refers to the lag between the command producing a physical event and the measured response from a sensor.

Enabling PAC causes the program to adjust the servovalve command signal until the programmed amplitude is achieved, and the phase lag is removed.

Time and frequency

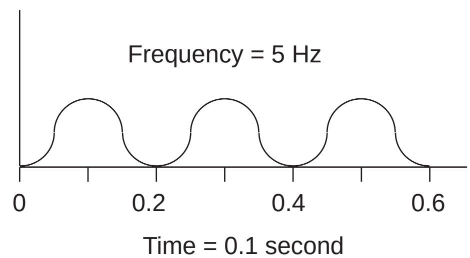
Selecting time allows you to specify the time for a single-segment cycle (half cycle). Selecting frequency allows you to specify the number of cycles per unit of time. A cycle consists of two single-segments.

TIME UNITS
milliseconds
seconds
hours
days

FREQUENCY UNITS
hertz (Hz)
cycles per second (cps)
cycles per minute (cpm)
cycles per day

Selecting time or frequency changes the rate type value label to Time or Frequency respectively. Press the list icon of the right rate type entry field to select the units of measurement.

The units for Time are based on a single-segment. The units for Frequency are based on a cycle (two segments).



Both rate types can specify the same waveform using different types of values.

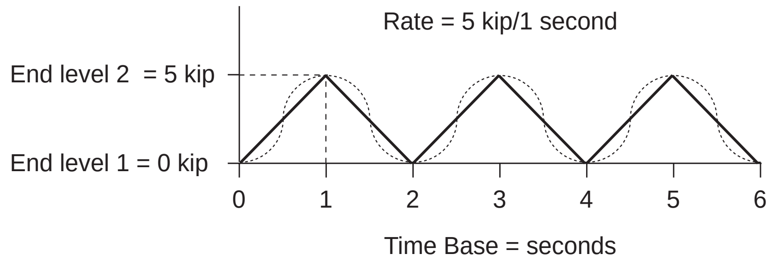
Rate

Selecting rate allows you to specify a segment with a constant rate between end level 1 and end level 2. A rate value represents the amount the control mode changes in one time unit.

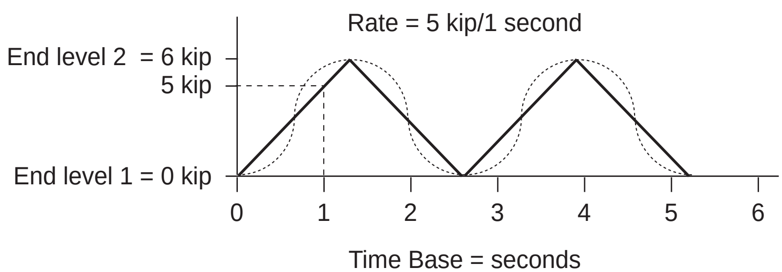
Rate is typically associated with a ramp. You specify the rate as if a ramp is being programmed. The segment shape of a haversine or step (square wave) is maintained when you use rate.

Selecting rate changes the rate type value label to Rate. Select the units of measurement to the right of the rate type value field.

A rate is expressed as units per time for a single segment.



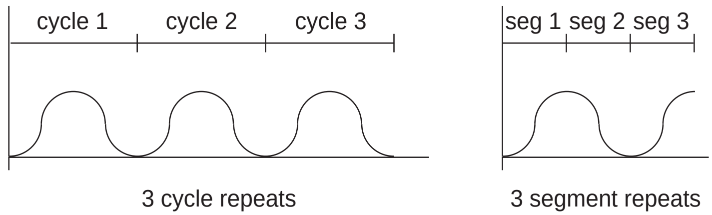
The rate is maintained even though the end level has changed.



Repeats

You can repeat cycles or segments. Two segments create one cycle. Repeating a full cycle starts at level 1 and ends at end level 2. Repeating a segment can end at end level 1 or end level 2.

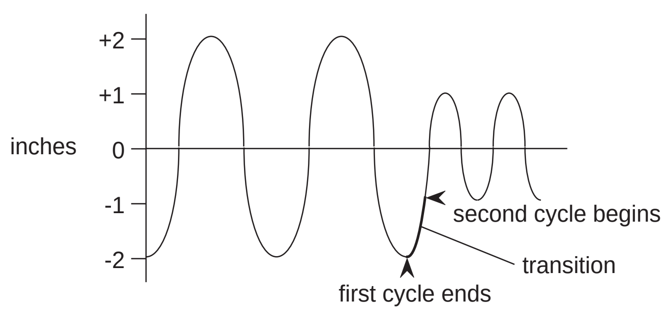
Entering 0 repeats causes the waveform to cycle continuously.



End levels

The two end levels specify the amplitude of the cyclic waveform. When two test command processes with different end levels are sequenced, the transition between them is accomplished at the rate and segment shape of the following segment. This provides a smooth transition between test commands.

For example, assume a cyclic command process with end levels -2 and +2 inches cycles twice. When the process is complete, the next cyclic process with end levels -1 and 1 inch begins at the -2-inch level and bridges -2 inches to -1 inch at its frequency.



Relative end levels

Relative end levels are available if you select a relative segment shape. Instead of specifying an absolute end level (as described in the example above), the end levels are relative to the start of the process. Using the same end levels as in the example above, if the process started with displacement a +1 inch, the command would cycle between -1 and +3 inches.

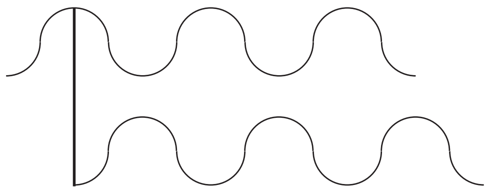
The Processes

Phase lag

The Phase Lag specification is available when the cyclic command is to be applied to more than one control channel. One control channel should have a 0° phase; this provides a reference for another control channel phase setting.

For example, assume two control channels use the same cyclic waveform. The axial control channel starts the waveform while the torsional control channel starts the waveform 180° after the axial waveform has started.

Axial control channel = 0° phase



Torsional control channel = 180° phase

Using the window

Some window selections affect other selections within the window. So, complete the selections in the order given.

1. Select a control channel.
2. Complete the control channel portion of the window to define cycle parameters that are unique for each control channel.
   - Select a control mode, end level units, and the phase units. Enter the two end levels and (if more than one control channel is listed) enter the phase value.
3. The upper portion of the window defines the cycle parameters that are common for each of the listed control channels.
   - Select a segment shape, rate type, rate type units, and the type of repeats. Enter the rate type value and the number of repeats.

Data Acquisition

A data acquisition process acquires data from any sensor signal. This type of process should be sequenced in parallel with a command process. Meaningful data cannot be acquired unless the control channel is doing something.

Note *Pages 230 through 233 provide details common to all processes such as default settings, using triggers, and how to reach the process windows.*

Contents

How it works

A data acquisition process can be configured in a variety of ways. The following attributes may be combined primarily to trigger other processes, to acquire data, or a combination of both.

- Four data modes to define how data is collected.
- Five buffer types to define how data is recorded.

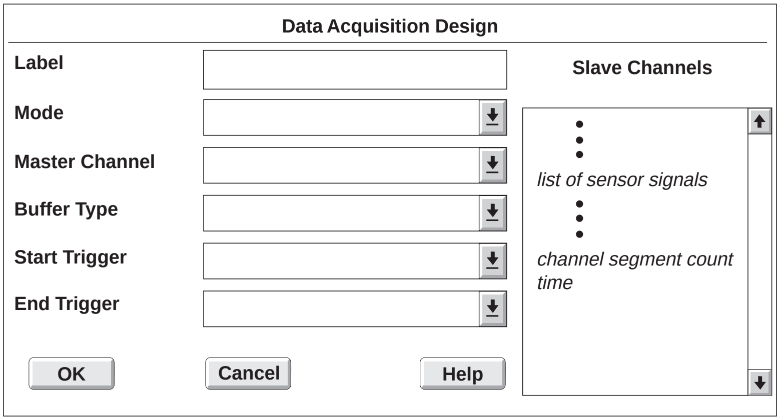
The process monitors a master channel according to the selected data mode. The master channel can be a sensor signal or time. A sensor signal can detect level changes or peaks/valleys. Time can acquire data from any slave channels at set intervals.

When the master channel detects the data mode requirement, data is acquired from the master channel and all slave channels. Each channel of data can have different units assigned to it. Data is acquired until a data buffer is full. A buffer's size and type determine how data is recorded and how it affects other processes.

Data Acquisition Design Window

The information in this window is saved with the test template.

The information in this window is saved with the test template.



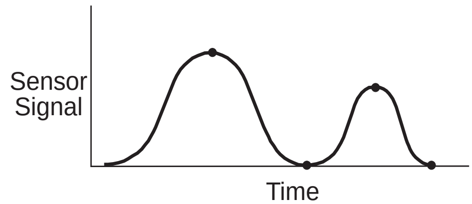
CONTROL	FUNCTION
Label	Names the process. Type the name you want to call the process in the entry field.
Mode	Selects the type of data to be acquired.
Master Channel	Selects the input signal that determines when data is recorded. Select a sensor signal or *Time*.
Buffer Type	Specifies how data is temporarily stored until the data is saved to disk.
Start Trigger	Specifies the beginning of the process. Press the list icon and select a trigger.
End Trigger	Specifies the end of the process. Press the list icon and select a trigger.
Slave Channels	Selects additional input signals for data acquisition. Highlight any input signal to select it as a slave channel. You can select as many slave channels as you wish.

Modes

The type of mode you select determines what kind of data is recorded. Each of these modes can be used to acquire certain data.

Peak/Valley Valley/Peak

Records data when the master channel signal detects a peak or valley. Setting a sensitivity value allows you to specify the amount of reversal that is necessary to define a peak or valley.

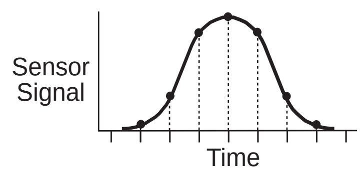


The peak/valley mode looks for a peak first whereas the valley/peak mode looks for a valley first. Otherwise, these modes are the same.

For example, you could acquire data every time a peak or valley is detected on a force channel and, at the same time, take data on the strain and displacement channels. Another way to use this mode is to count cycles to trigger another data acquisition process.

Timed

Records data at specified equal time intervals.

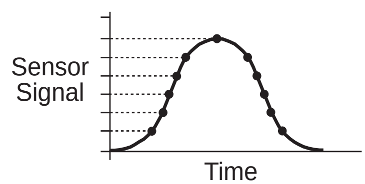


Time becomes the master channel.

For example, data could be acquired once each second on all selected data channels.

Level crossing

Records data when the master input signal changes a specific amount. In this case, data samples are taken on all desired channels each time the master signal acquires data.



For example, data could be acquired from the force, strain, and displacement sensors each time that force (selected as the master channel) changes 500 newtons.

The Processes

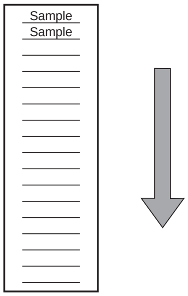
Buffer type

Because each type of buffer offers different characteristics, you may define one data acquisition process to simply trigger another data acquisition process to acquire data.

Single buffer

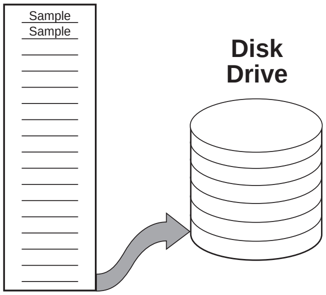
Data is recorded to fill the buffer once, then stops the process and saves the data to disk. The size of the data buffer determines how much data to collect.

This selection acquires specific data in a test, and is useful for triggering other processes.



Continuous buffer

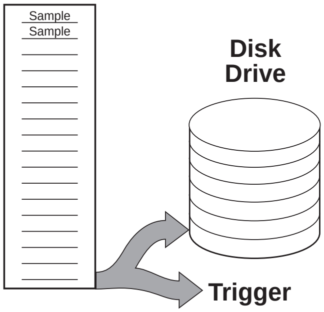
In this mode, data is continuously taken and automatically stored to disk. Storing continues until the end trigger requirement occurs or the step done process is encountered. The only limit to the total number of samples is the drive's storage capacity.



Continuous with trigger buffer

This buffer type functions the same as the *continuous* buffer except it issues a trigger each time the buffer if full. You specify the size of the buffer to determine when the trigger is issued. This process ends at the end of a step.

This selection is useful to trigger other processes at regular intervals while saving data.



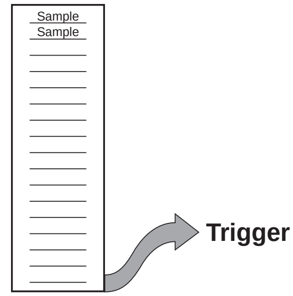
Continued…

Buffer type (continued)

Trigger only buffer

This buffer type functions the same as the *continuous with trigger* buffer except it does not save data. You specify the size of the buffer to determine when the trigger is issued. This process only ends at the end of a step.

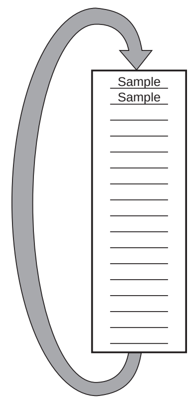
This selection is useful to trigger other processes at regular intervals without saving data.



Circular buffer

A circular buffer continuously records data to the buffer. When the buffer is full, the data is loaded into the top again, overwriting the oldest data. This continues until there is an end trigger or the step ends and saves the data to disk.

This type of buffering is useful when the data just before some event (such as specimen failure) is crucial, but data is not required for the whole test.



Slave channels

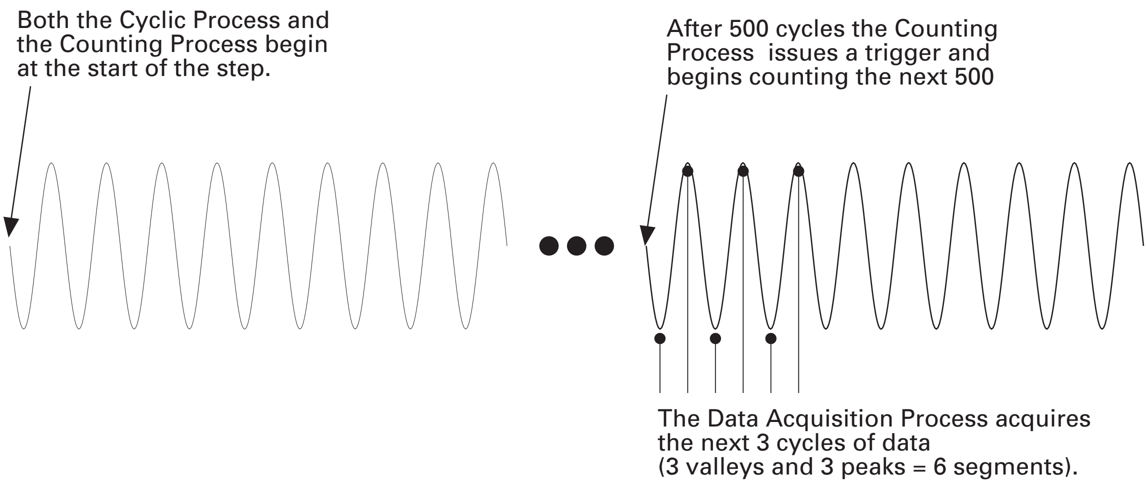
Each time the master channel acquires data (according to the selected mode), the slave channels also record data to the buffer.

- A segment count slave channel records the current number of segments that have been run.
- A time slave channel records the amount of time since the test began.
- An input signal slave channel records the current signal level of the sensor.

Trigger example

Suppose you want to acquire cycles of data every 500 cycles. In this case you can use one data acquisition process to count the cycles and trigger another data acquisition process to acquire data. If the process that counts cycles uses a trigger only buffer, it will be able run the other data acquisition process multiple times. Review the process attributes in the following table.

Cyclic Process	Counting Process	Data Acquisition Process
start trigger = step start	start trigger = step start	start trigger = Counting Process
end trigger = <none>	end trigger = <none>	end trigger = <none>
segment shape = haversine	mode = level crossing	mode = valley/peak
frequency = 2 Hz	buffer type = trigger only	buffer type = single
repeats = 3003	master channel =axial segments	master channel = force
control mode = force	data header = count segments	data header = 3 cycles of data
endlevel 1 = -500 lbf	level increment = 500 cycles	sensitivity = 100 lbf
endlevel 2 = -1000 lbf	buffer size = 1	buffer size = 6



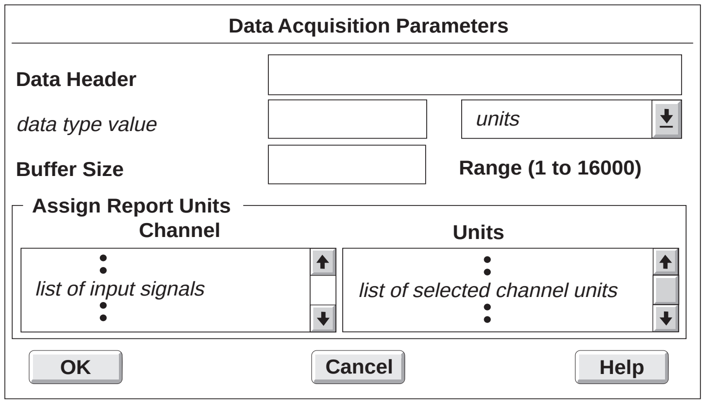
Unlike other processes, the counting process can issue more than one trigger. Since the Data Acquisition Process starts every time the Counting Process issues a trigger, it is able to run more than once.

Data Acquisition Parameters window

You must establish a data acquisition process with the Data Acquisition Design window before you can use this window. The information in this window is saved with the test procedure.

.

Use this window to define the parameters of the process



Control	Function
Data Header	Labels the data in the data file.
data type value	Specifies how data is acquired. The label for this field depends on the Mode you selected in the Data Acquisition Design window. **Label** — **Mode** Level Increment — Level Crossing Time Increment — Time Sensitivity — Peak/Valley Valley/Peak Type the value you want to specify for the data acquisition requirement in the left entry field.
Buffer Size	Specifies the number of data elements the buffer can store. A data element represents one data sample of the master channel and each slave channel.
Assign Report Units	This area of the window specifies the units for each master and slave channel. Select a channel, then select the units for that channel. Repeat this for each channel.

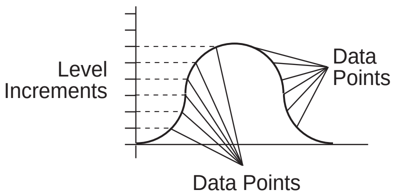
Buffer size

The default buffer size is 1024. You can set the data buffer to record 1 to 16,000 data elements. A data element includes the data from the master and slave channels along with a time stamp.

Buffer sizes below 64 cause extra traffic on the communication lines between the computer and digital controller. Acquiring data at fast rates and saving that data can cause the computer to become sluggish (slow to respond to selections). Tests that acquire data at fast rates can benefit from larger buffer sizes.

Level increment

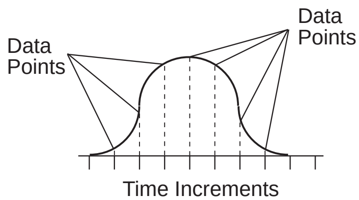
A level increment value specifies how much the master channel signal changes between data acquisitions. The increment level is referenced to zero and not from the beginning value of the process.



For example, assume you want to acquire data every time the displacement changes 2 millimeters. The master channel must be a displacement (or length) input signal. Select the master channel and millimeters units in the Assign Report Units area. Enter 2 in the Level Increment entry field. When the master channel reaches 2, 4, 6, etc. millimeters, data is acquired for the master channel and all slave channels.

Time increment

A Time increment value specifies how long between data acquisitions.



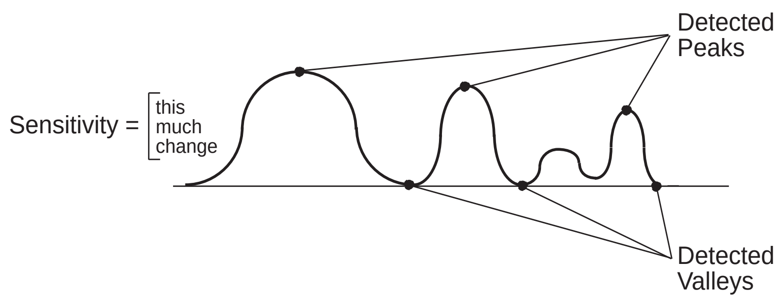
For example, assume you want to acquire data once each second. The master channel is always called time for this mode. Select the time channel and seconds for units in the Assign Report Units area. Enter 1 second in the Time Increment entry field. Data is acquired for all slave channels each second.

The Processes

Sensitivity

A sensitivity value specifies how much the master channel feedback signal must change to detect a peak or valley. The highest or lowest value is remembered (along with slave channel data) until the master feedback signal changes by the amount of the sensitivity value, then the peak or valley data is recorded.

Setting the sensitivity too low may cause signal noise to be recognized as peaks and valleys. Setting the sensitivity too high may cause low-amplitude signals to be missed.



For example; assume you want to acquire data every time a peak or valley is detected on a force channel. Also assume you expect the force channel to operate within 0 to 6 kN and you want data for peaks and valleys that exceed 2.5 kN.

Select a force input signal as the master channel. Select the master channel and kN units in the Assign Report Units area. Enter 2.5 in the Sensitivity entry field. Each time the master channel changes more than 2.5 kN, a peak or valley is detected. Data is acquired for the master channel and all slave channels.

Assigning report units

Use this procedure to assign units to each master and slave channel.

1. Highlight a channel in the Channel column. This causes the Units column to display the appropriate units for the dimension of the channel.
2. Highlight the units you want assigned to the channel data.
3. Repeat this procedure for each channel.

Data Files

A data file contains the data acquired from a test along with a label indicating the type of data and the units of the data. Data can be acquired for all available input signals. The data file format can be selected for use with popular spreadsheet programs or plain text.

Below is an example of the output from a data file.

Data File Format

This shows the types of information that can be saved in a data file.

The data format is in ASCII.

The file contains tab separators so that the columns of data automatically appear in separate columns in a spreadsheet program.

Test Control information is included when Log Events is enabled.

You can type a test description.

Test Control	Started	Time: 0.750

Acquired data every 0.1 second for the displacement channel.
Specimen Number 9669A. Operator--Scout Johnson

DRP Process Data	simple test	Time: 11.904
Time	Length 1	
Sec	mm	
1.331	-9.31775	
1.431	-9.32211	
1.532	-9.31775	
1.632	-9.31775	
1.732	-9.32211	
1.832	-9.32211	
1.932	-9.32211	

Each buffer of data from a Data Reduction Process (DRP) is headed with a label you provide.

Any number of input channels of test data

Operator Note Time: 3.908
First visible sign of deformation

Test Control	Done	Time: 12.816

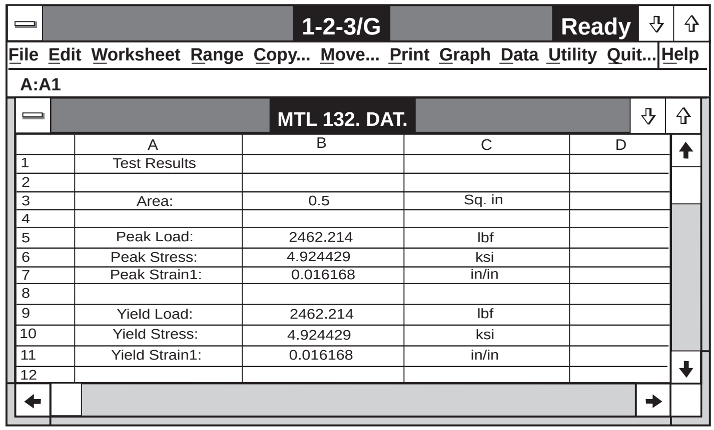
You can add an Operator Note any time during a test.

Using data acquisition for analysis

After the data has been gathered, it may be processed by analysis, plotting, or word processing programs. Below are examples of data processed by two well-known spreadsheet programs.

Lotus 1-2-3 Window

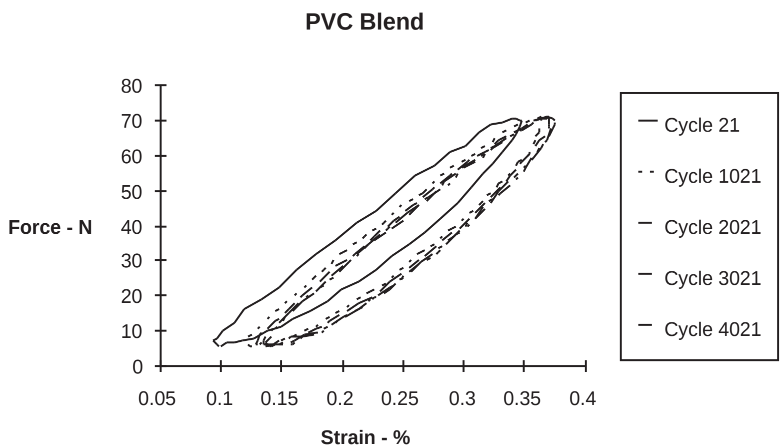
Here is an example of tensile test data on a composite material as analyzed by Lotus 1-2-3.



	A	B	C	D
1	Test Results			
2				
3	Area:	0.5	Sq. in	
4				
5	Peak Load:	2462.214	lbf	
6	Peak Stress:	4.924429	ksi	
7	Peak Strain1:	0.016168	in/in	
8				
9	Yield Load:	2462.214	lbf	
10	Yield Stress:	4.924429	ksi	
11	Yield Strain1:	0.016168	in/in	
12				

Excel Plot

Here is an example of a set of hysteresis loops that have been plotted using Microsoft Excel.



Data Limit Detector

This type of process produces an event and should be sequenced in parallel with a command process. Meaningful data cannot be acquired unless the control channel is doing something.

Note *Pages 230 through 233 provide details common to all processes such as default settings, using triggers, and how to reach the process windows.*

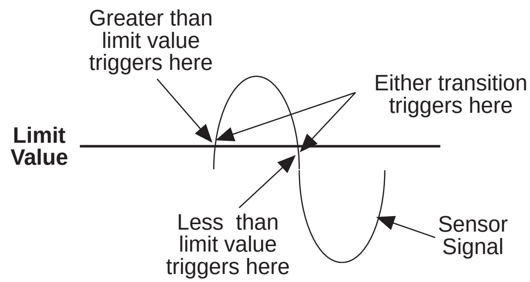
Contents

How it works

The data limit process is useful to trigger other processes. A limit can be a number segments, an amount of time, or a sensor signal value. A limit can be specified as a relative value (starting when the process begins) or an absolute value (from a zero reference). Three types of trigger options offer different ways to complete the process.

For example, a data limit detector can monitor an input signal for a specific value. The process compares the current value to the limit value.

According to the trigger options the process can end if the current value is more than, less than, or crossing the limit value.

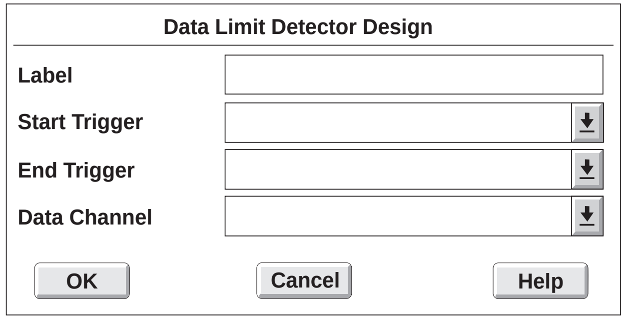


Note *Don't use this process to protect equipment. Instead, use the TestStar detectors because they are faster. This process is intended to trigger other processes.*

Data Limit Detector Design Window

This window names the process and specifies when the process starts and stops.

The information in this window is saved with the test template.



CONTROL	FUNCTION
Label	Names the process. Type the name you want to call the process in the entry field.
Start Trigger	Specifies the beginning of the process. Press the list icon and select a trigger.
End Trigger	Specifies the end of the process. Press the list icon and select a trigger.
Data Channel	Selects an input signal, time, or completed segments of a control channel for the data limit to monitor.

Data channel

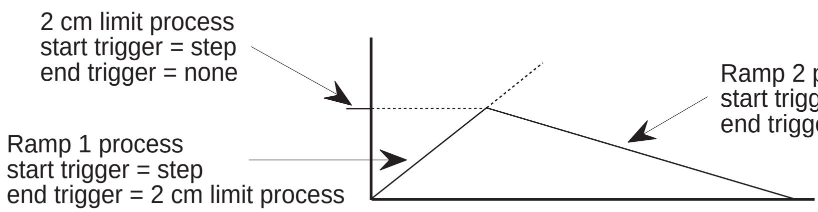
The data channel for the data limit detector determines how the limit is detected. There are three types of selections:

- **Time** – specifies how long the process waits.
- **Control channel segments** – counts the number of segments of the selected control channel. All defined control channels are listed.
- **Input signal** – monitors a sensor signal. All defined input signals are listed.

Trigger example

Assume you want to stop a ramping process when sensor reaches a specific level, and then you want to start a second ramp. You can create a data limit detector process by specifying the following attributes:

- the type of sensor feedback you want to monitor
- the limit level where you want to stop the ramp
- an absolute limit level
- the trigger option as either transition (the process triggers when it crosses the limit level)

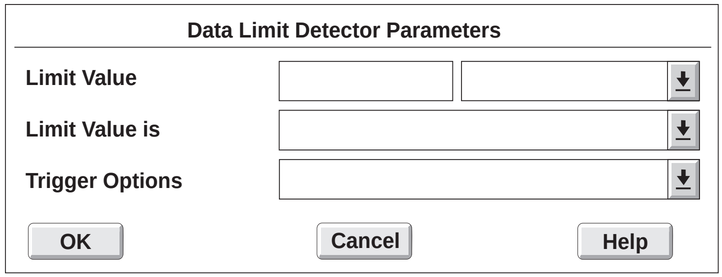


For example, the test command ramps until the 2 cm limit process detects the sensor feedback has reached its limit level (in this case, 2 cm). The completion of the data limit detector process triggers the end of the ramp 1 process. When the ramp 1 process ends, the ramp 2 process begins.

Data Limit Detector Parameters Window

You must create a data limit detector process in a template with the design window before you can access this window. The information in this window is saved with the test procedure.

Use this window to define the parameters of the process.



CONTROL	FUNCTION
Limit Value	Specifies a value to end the process. Pressing the list icon shows the available units for the channel specified in the design window.
Limit Value is	Specifies whether the limit value is absolute or relative.
Trigger Options	Specifies if the limit value is detected when it is greater than, less than, or crosses the limit value.

Limit value

The limit value can be expressed one of three ways (according to the channel selection in the Data Limit Detector Design window).

- **Time** is expressed with a unit such as hours, minutes, or seconds.
- **Segments** can be counted for the selected control channel. You can count either segments or cycles. Each cycle of a cyclic command consists of two segments. Counting cycles should be selected only when monitoring a cyclic command. Counting segments works for all command processes.
- **Sensor** feedback signals can be monitored to detect a specific sensor value or a change in sensor value.

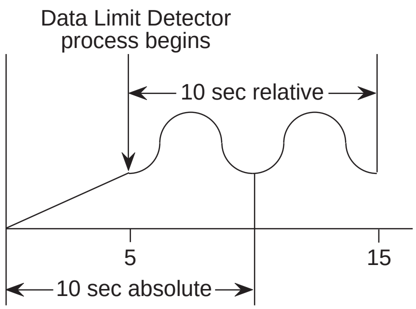
Time and segment values

The limit value can be specified as an absolute or relative value.

The absolute selection for the time and segment count limit values reference the beginning of the test. The relative selection references the limit value to the beginning of the process.

For example, assume the data limit detection process begins after a 5-second ramp.

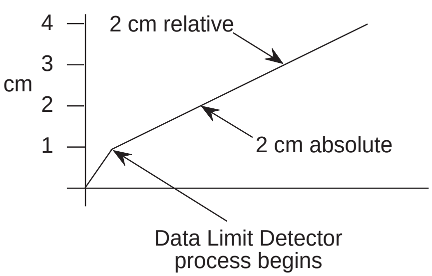
- An absolute time limit of 10 seconds causes the data limit detector process to end 5 seconds after it starts (10 seconds after the test begins).
- A relative time limit of 10 seconds causes the data limit detector process to end 10 seconds after it starts (15 seconds after the test begins).



Level values

The absolute selection for a sensor limit value references the value to zero. The relative selection references the limit value as a change from whatever the sensor feedback value may be when the process begins.

For example, assume the test command ramps from 1 cm to 4 cm. An absolute limit value of 2 cm ends the process at 2 cm. A relative limit ends the process at 3 cm.



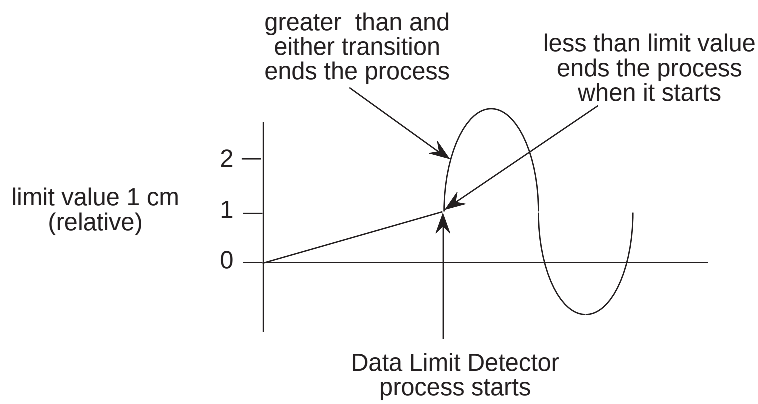
The Processes

Trigger options

A limit can be detected in one of three ways.

- Selecting **Greater than limit value** causes the process to end whenever the selected sensor feedback is more positive than the limit value.
- Selecting **Less than limit value** causes the process to end whenever the selected sensor feedback is more negative than the limit value.
- Selecting **Either transition** causes the process to end whenever it crosses the limit value.

For example, assume a relative data limit value of 1 cm. The *Greater than limit value* and *Either transition* selections are detected when the sensor feedback exceeds 2 cm. The *less than limit value* selection requirements are met as soon as the process begins because the output is already below the limit value.



Digital Input Detector

This type of process produces an event and should be sequenced in parallel with a command process. Use this process to trigger other processes when an external device issues a signal.

Note *Pages 230 through 233 provide details common to all processes such as default settings, using triggers, and how to reach the process windows.*

Contents

Prerequisites

Using a digital input detector process requires the following:

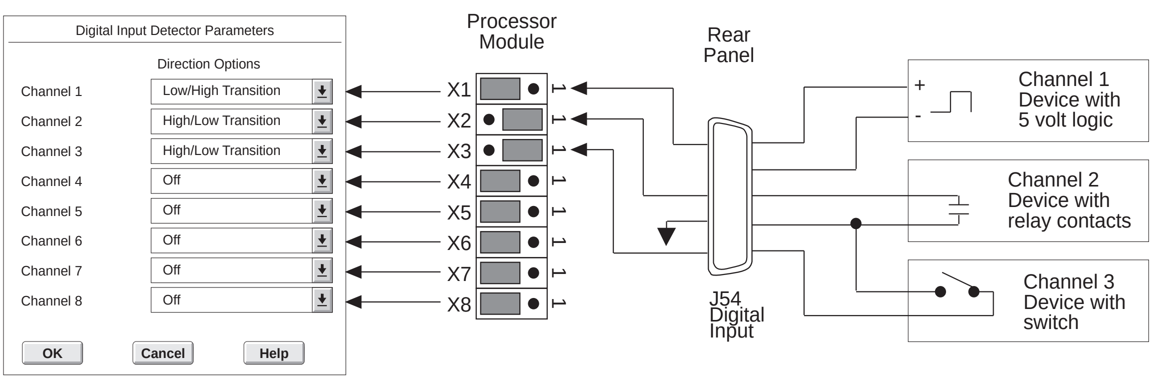
- Proper cabling from the external devices to connector J54 (see Chapter 3 in the TestStar Installation manual).
- Jumpers X1 - X8 on the Model 490.50 Processor Module must be configured for the each type of input signal (see Chapter 2 in the TestStar Installation manual).

How it works

A digital input detector process monitors up to 8 digital inputs from the rear panel connector J54 (see Chapter 3 in the TestStar Installation manual). The cable may have more than one destination. The process ends when the proper signal is detected by any of the eight input signals.

- The the jumpers on the processor module can be configured to detect logic signals or relay contacts.
- The process determines the type of signal that is expected by each input. The types of signals that can be detected are:

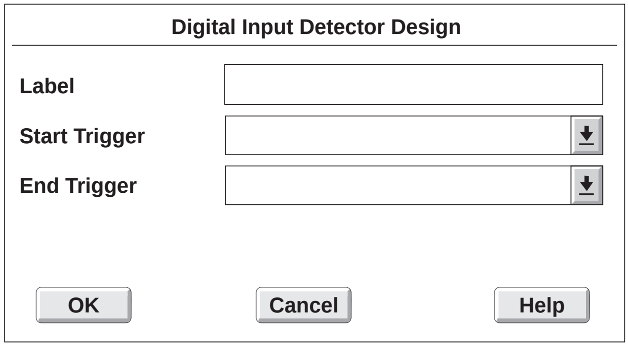
Logic Signal	Contacts (switch or relay)
low/high transition	switch opens
high/low transition	switch closes
either transition	switch opens or closes
low	switch is closed
high	switch is open



Digital Input Detector Design Window

This window names the process and specifies when the process starts and ends.

The information in this window is saved with the test template.

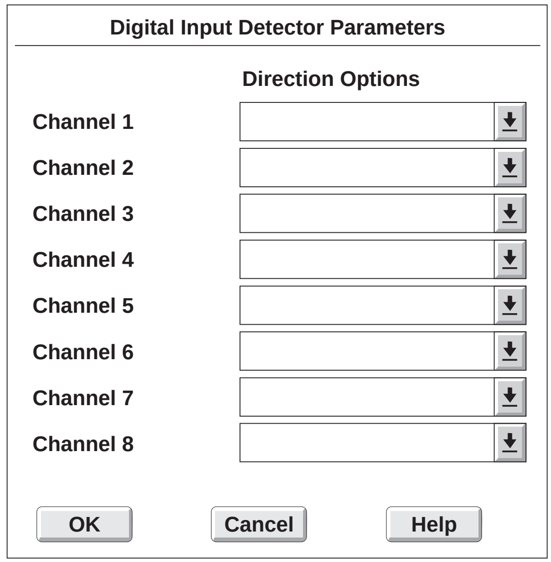


Control	Function
Label	Names the process. Type the name you want to call the process in the entry field.
Start Trigger	Specifies the beginning of the process. Press the list icon and select a trigger.
End Trigger	Specifies the end of the process. Press the list icon and select a trigger.

Digital Input Detector Parameters Window

You must create a data input detector process in a template with the design window before you can access this window. The information in this window is saved with the test procedure.

Use this window to establish the detector requirements for each of the 8 digital inputs from the rear panel connector J54.



CONTROL	FUNCTION
Channel	Each channel corresponds with one of the eight inputs through the rear panel connector J54. When you use the digital input detector you must know what devices are connected to each channel.
Direction Options	Each channel can become active with different signal transitions. You should know how the external device becomes active to set its channel transition. ◆ Low/High Transition — switch opens ◆ High/Low Transition — switch closes ◆ Either Transition — switch opens or closes ◆ Channel Low — switch is closed ◆ Channel High — switch is open

Channel signals

You must know which device is connected to each channel. Refer to the Cabling chapter in the TestStar Installation manual for the connections to J54.

You must also know how each channel input becomes active so you can configure the processor module jumpers correctly Each channel of the digital input detector can be configured to monitor a logic signal or relay contacts. Jumpers X1 - X8 on the Model 490.50 Processor module configures each of the 8 inputs for a logic or contact type of signal. Each jumper corresponds with the digital input signal of the same number. Refer to the Hardware Installation chapter in the TestStar Installation manual to set the jumpers.

After the hardware is configured properly you can select the appropriate direction option for each channel.

Digital Output

This type of process produces an output to control external devices and should be sequenced in parallel with a command process. Use this process when you want to trigger an external device.

Note *Pages 230 through 233 provide details common to all processes such as default settings, using triggers, and how to reach the process windows.*

Contents

Prerequisite

Equipment monitoring the digital output signals should be connected to rear panel connector J55. You must know what type of signal will activate the external device (see Chapter 3 in the TestStar Installation manual).

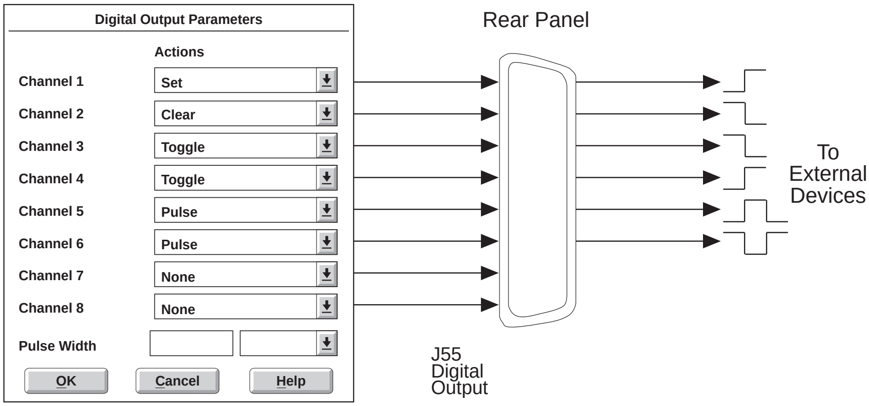
How it works

The Digital Output process can signal up to eight external devices during a test. The process produces up to 8 output signals through the rear panel connector J55.

For example, assume you have six devices you want to signal. Before the digital output is issued, the output for channels 1, 4, and 5 is +24 volts while the output for channels 2, 3, and 6 is 0 volts. When the process triggers, the outputs are issued.

All output channels that have an action assigned are issued when the process is executed.

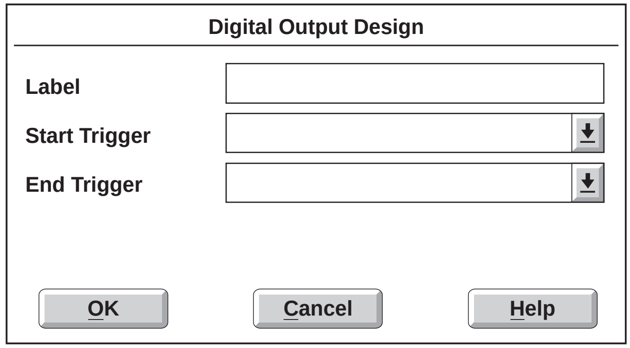
Channels set to None are disabled.



Digital Output Design Window

This window names the process and specifies when the process starts and stops.

The information in this window is saved with the test template.

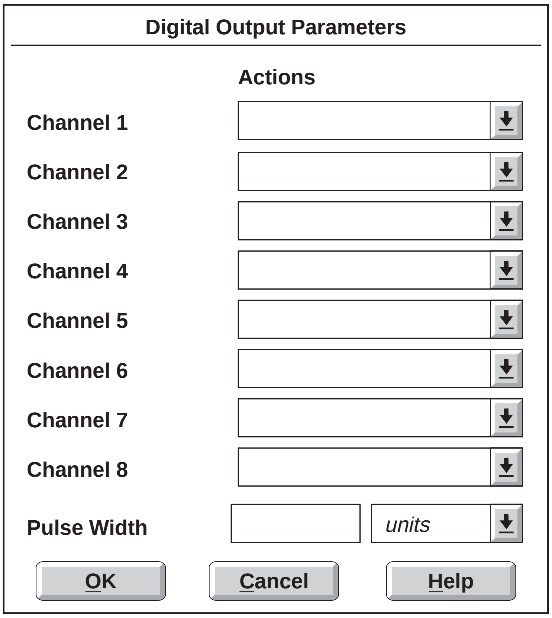


CONTROL	FUNCTION
Label	Names the process. Type the name you want to call the process in the entry field.
Start Trigger	Specifies the beginning of the process. Press the list icon and select a trigger.
End Trigger	Specifies the end of the process. Press the list icon and select a trigger.

Digital Output Parameters Window

You must create a digital output process in a template with the design window before you can access this window. The information in this window is saved with the test procedure.

Use this window to establish the output requirements for each of the 8 digital outputs from the rear panel connector J55.



Control	Function
Channel	Connector J55 on the digital controller provides up to 8 channels of digital output. When you use a digital output, you must know what digital devices are connected to that channel.
Actions	Set each channel to one of the following signals: ◆ None ◆ Set ◆ Clear ◆ Toggle ◆ Pulse
Pulse Width	Specifies the duration of a pulse action. Enter a value in the left entry field. The pulse width applies to all channels with the Pulse action selected.

Actions

The following define the actions that can be assigned to each channel.

Action	Definition
None	Disables the channel.
Set	Turns the channel on – a logic high signal (+24 Vdc).
Clear	Turns the channel off – a logic low signal (0 Vdc).
Toggle	Inverts the current state (from high-to-low or low-to-high).
Pulse	Inverts the current state for the duration of the Pulse Width specification.

Using an output

The digital output provides 24-volt signals. Review the following:

- Select an action that is appropriate for the device connected to the channel.
- If a pulse action is selected, enter a pulse width appropriate for the device.
- The 8 channels can be output to multiple destinations.
- All 8 outputs are activated when the process begins.
- When the process is complete, the outputs remain in the designated state. Only the pulse action returns the output to its original state.

External Command

An external command process uses a signal from an external device to control a servovalve or servo motor. This process should be sequenced in series with other command processes.

Note *Pages 230 through 233 provide details common to all processes such as default settings, using triggers, and how to reach the process windows.*

Contents

Prerequisites

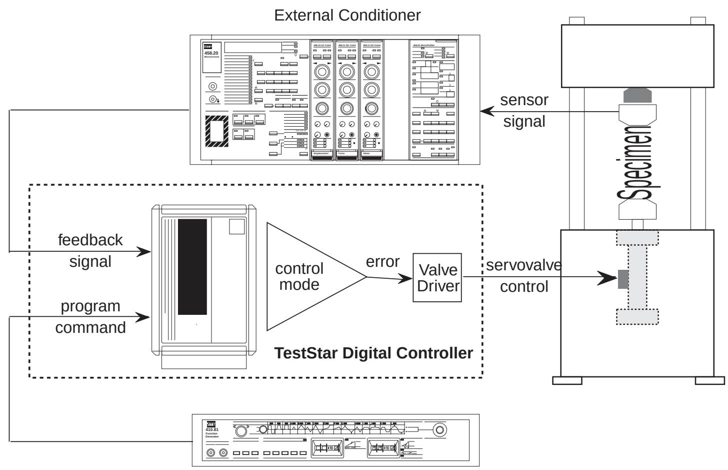
You must perform the following before you can use this process:

- Connect the command source to the digital controller
- Define the command source input signal and control mode
- Set up a temporary sensor to scale the input.

See Chapter 10 in the TestStar Installation manual for a procedure that defines an external signal for use with an external command process.

How it works

This process lets you use a command source connected to the rear panel of the digital controller to command the servo loop.

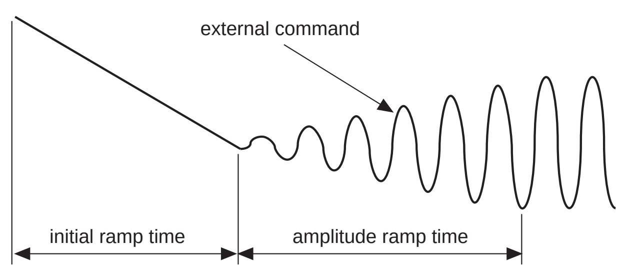


For example, assume you have an old command program that you don't want to reprogram with TestWare-SX. Also assume your old command program is a simple cyclic waveform (because it is easier to illustrate how it works).

You must start the external command source before your test reaches the external command process.

Since the external command is at an unknown level, the amplitude ramp ensures that you achieve smooth transition from the current command to the external command.

When the external command process begins, it ramps to the mean level setting of the process. Then it ramps the amplitude to full scale (or a percentage of it).



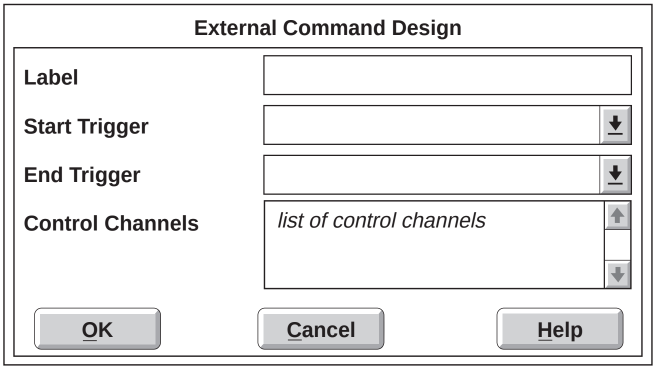
Once an external command process begins it will not end until another process triggers it or you press the Stop or Hold button. Unlike other command processes, the external command process cannot be setup to run for a specific number of cycles or a period of time.

External Command Design Window

This window names the process and specifies when the process starts and stops.

Note *Starting two test commands at the same time for the same control channel will display a message box that describes an error condition.*

The information in this window is saved with the test template.



CONTROL	FUNCTION
Label	Names the process. Type the name you want to call the process in the entry field.
Start Trigger	Specifies the beginning of the process. Press the list icon and select a trigger.
End Trigger	Specifies the end of the process. Press the list icon and select a trigger.
Control Channels	Specifies which control channels the process is to be applied (more than one may be selected). A list of the available control channels is shown.

CAUTION

Your test will appear to hang up if the source of the external command is not running.

Be sure that the external device that produces the external command is running when you start this process.

Trigger example

Assume you have a test profile that you created on another program source and you don't want to reprogram it into a TestWare-SX template. Also assume you have completed the prerequisites for this process.

Review the following to **start** the external command process:

- If the external command process is the first command process of your procedure, use the step start trigger. Be sure the program source is running before you start your test.
- You could create an operator event that prompts you to start the program source.
- If your command source can accept a remote start signal, you could use a digital output process to start the external command source. The digital output process could also start the external command process.

Review the following to **end** the external command process:

Note *You REALLY need to know what the external program does so you can determine how to stop it (either a detection process or an interlock).*

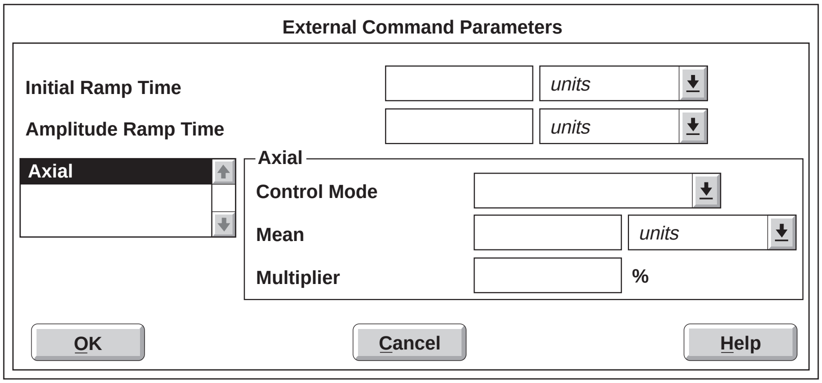
- You could use a data limit detector to stop the external command after a specific amount of time.
- You could use a sequence of data limit detectors that detect specific levels and/or time (according to the nature of your command).
- If your command source can issue a signal when the profile is complete, you could use a digital input process to stop the process.
- You can configure the TestStar detectors (error, under peak, and limits) to stop the program and ramp or hold the command (see the TestStar manuals). Use the <none> trigger for this application.
- You can also manually stop the process (and procedure) with the Stop button in the Execute window or on the load unit control panel. Use the <none> trigger for this application.

The Processes

External Command Parameters Window

You must create an external command process in a template with the design window before you can access this window. The information in this window is saved with the test procedure.

Use this window to integrate the external command with TestWare-SX.

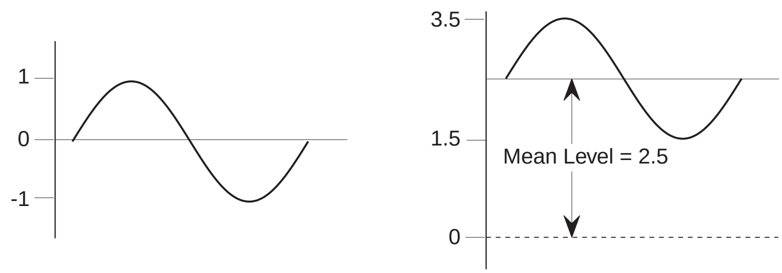


CONTROL	FUNCTION
Initial Ramp Time	Specifies the amount of time the command ramps from starting level of the process to the mean level setting. The ramp is expressed with units of time.
Amplitude Ramp Time	Specifies the amount of time the program amplitude ramps from zero to full-scale The ramp rate value represents units per second.
control channel	Specifies the control channel. For a single-channel system, one choice is available (typically called Axial). Each control channel can have different parameters.
Control Mode	Specifies the control mode for the process. Pressing the list icon shows the control modes available for the selected control channel.
Mean	Sets the mean level of the external command
Multiplier	Specifies a multiplication factor to scale the incoming signal.

Initial ramp to mean

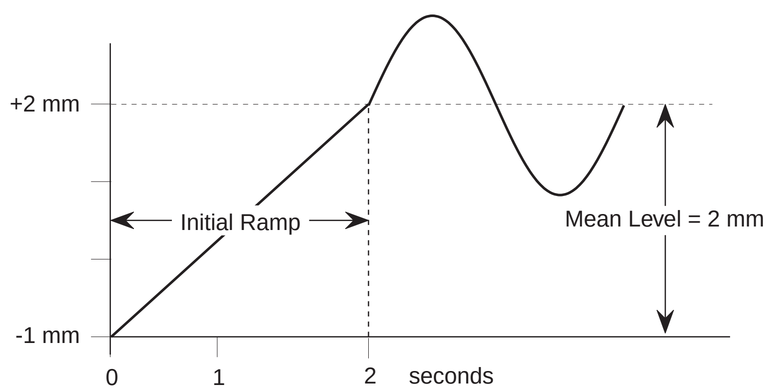
The Mean value introduces an offset that references the command to a level other than zero. The mean level value units are appropriate for the selected control mode. Adjust the Mean level control within ±full-scale of the control mode.

For example: suppose you want to run a 2 cm sine waveform between 1.5 and 3.5 cm. Adjust the Mean level control for 2.5 cm.



The external command begins with a ramp from the starting level of the process to the mean level of the process The initial ramp specifies the amount of time to execute the ramp. Once the initial ramp is complete, the amplitude ramp can begin.

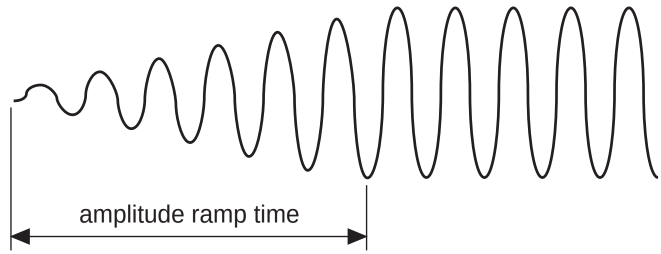
For example; suppose the starting level is -1 cm, the mean level is +2 cm, and the ramp time is 2 seconds.



Amplitude ramp time

The Amplitude Ramp Time allows you to slowly apply the program command to the specimen when the program starts. This feature is also called soft start and it prevents sudden actuator movement when the external command starts.

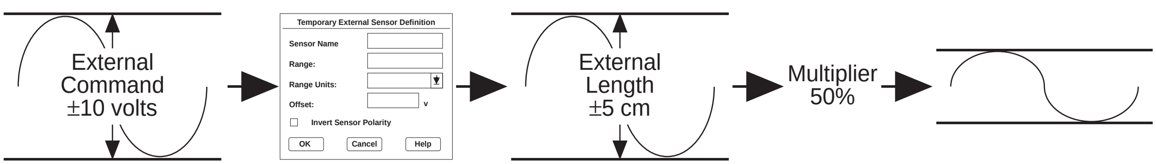
A command begins with an amplitude of zero and gradually increases until the programmed amplitude is reached.



Multiplier

The Multiplier control establishes the span (peak-to-peak amplitude) of the external command. The multiplier value scales the external command signal as a percentage of full-scale.

The Temporary External Sensor Definition window defines the maximum external command signal (±10 volts) in terms of engineering units.



For example, assume you have an external command signal that produces ±10 volts at full scale which represents ±5 cm and you want to use this signal for a ±2.5 cm test.

File Playback

A file playback process produces a command signal that controls a servovalve or servo motor. This process should be sequenced in series with other command processes.

Note *Pages 230 through 233 provide details common to all processes such as default settings, using triggers, and how to reach the process windows.*

Contents

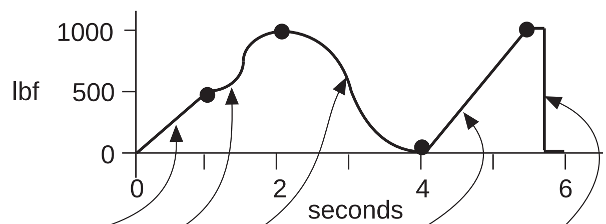
Prerequisites

You must create a playback file for use with the process. You should also be familiar with the type of information in the file.

How it works

A file playback process is a command process that reads a file for segment data (profiles, block data). From the segment data, the file playback process produces command segments to control the servovalve. The process uses files that you create with an application, such as a text editor (but not the OS/2 editor), a spreadsheet application, or an application designed to create profiles. To work with the file playback process within TestWare-SX, the data in the files must be formatted in a special way, see ***File Format*** on page 294.

In this example, each segment command identifies a waveform shape, a rate type, and a segment end level. The file is read in sequence to produce a waveform.



Shape	Time sec	Level lbf
ramp	1	500
sine	1	1000
sine	2	0
ramp	1.5	1000
square	1	0

The content of a playback file consists of a series of command elements which are defined in rows. Each command element can define a single segment (as shown above). Each segment is defined with a waveshape, rate or time or frequency, and end levels. A single playback file can (and often does) contain the command content of an entire test procedure.

The end level data can be entered with common units or normalized units. When a playback file is selected, it is scanned to determine what kind of units are appropriate for the test parameters. Normalized units can be expressed as a percentage of full-scale or multiplier values.

Continued…

How it works (continued)

Waveform compensation can be disabled, manually scaled or automatically optimized using a spectrum amplitude control (SAC) algorithm.

- Manual compensation is accomplished with parameters that adjusts the amplitude, and mean level. A runtime compensation window allows you to manually adjust the mean level and amplitude of the waveform.
- SAC compensation is accomplished with an algorithm that uses a table while the test is running to keep track of end levels. Each command end level is compared to its feedback end level to determine if an over programmed level is needed. Each time the file is run, new compensation values can be established or existing values can be used.

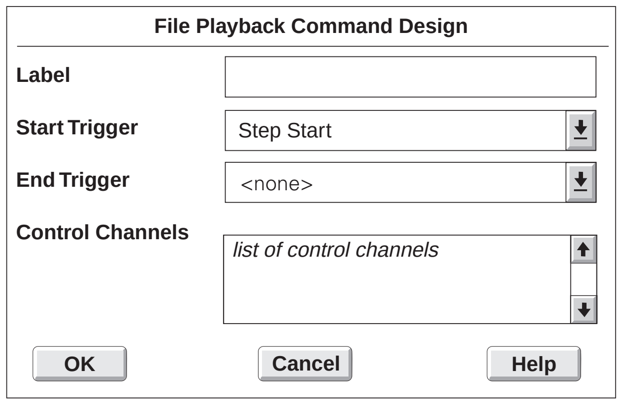
Note *In the event of a test recovery operation, you will need to re-establish the setting of the runtime windows (manual or SAC).*

File Playback Design Window

This window specifies when the process starts and stops. The information in this window is saved with the test template.

Note *Starting two test commands at the same time for the same control channel will display a message box that describes an error condition.*

Use this window to create a process that reads an ASCII file to control the test command.

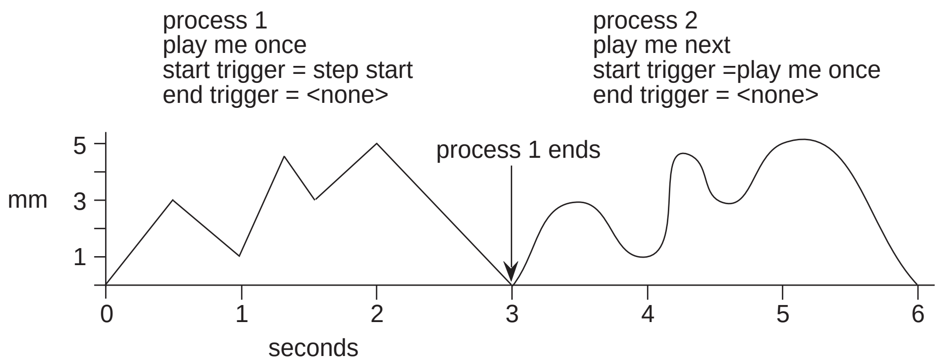


CONTROL	FUNCTION
Label	Names the process. Type the name you want to call the process in the entry field.
Start Trigger	Specifies the beginning of the process. Press the list icon and select a trigger.
End Trigger	Specifies the end of the process. Press the list icon and select a trigger.
Control Channels	Selects the control channels that the process is to be applied (more than one may be selected).

Trigger example

Assume you create a step with two file playback command processes. The only difference between the two file playback command processes is the segment shape. Process 2 is a copy of process 1 with *shape constant* changed (the file is assigned a waveform).

When the <none> trigger is used, the process continues until the entire playback file has been executed.



File Playback Parameters Window

The information in this window is saved with the test procedure.

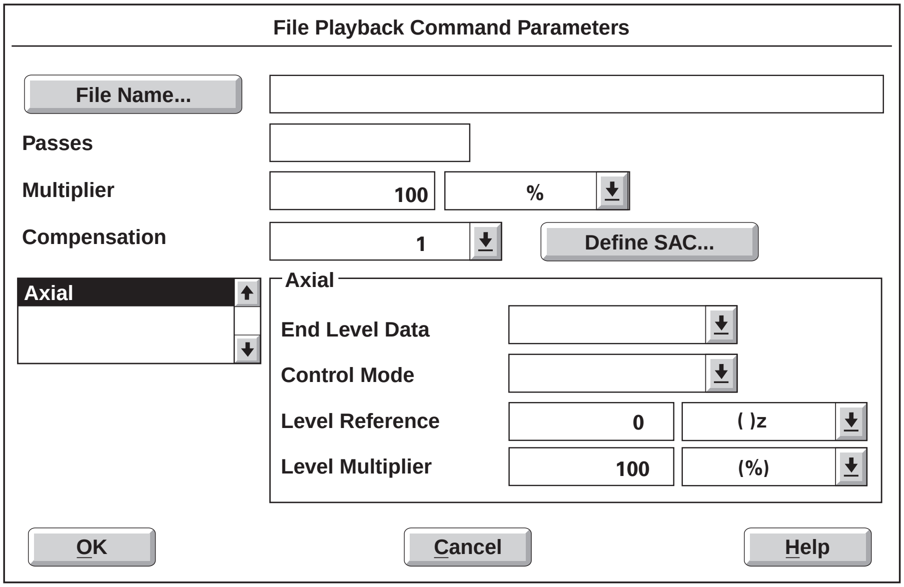
Prerequisite

You must create a file playback process in a template with the File Playback Design window before you can run this process.

TestStar versions 3.0 and later require the Model 490.50B or 490.50C Processor board.

Note *The* ***Define SAC…*** *button does not appear if an older version of the Model 490.50 Processor module is installed.*

Use this window to select a playback file and scale the file data.



Control	Function
File Name	Pressing the File Name button displays the Select end level file window where you select the file for this process.
Passes	Type the number of times the file is repeated. Enter 0 for continuous repeats; enter 1 for a single pass.
Multiplier	Scales the time component of the waveform with a multiplier using the units of the rate type of the playback file.
Compensation	Selects manual or SAC compensation of the waveform. Select none for normal operation. The manual selection displays a run time window where you can adjust the amplitude and mean level of the waveform during the test. The SAC selection enables the Define SAC button where you set up the SAC option.
Define SAC	Displays the Define SAC Compensation Parameters window where you select a spectrum amplitude control table, table limit, and error tolerance.
control channels	Displays a list of the available control channels. The playback file can be applied to more than one control channel. Selecting a control channel displays its name to define the channel for the end level data, control mode, and level parameters.
End Level Data	Selects the end level data. Each playback file can have up to four columns of end level data. End level data is a list of values that specify the end level for each segment in the playback file.
Control Mode	Displays a list of the available control modes. The playback file determines which control modes are valid for the selected end level data.
Level Reference	Specifies a mean level offset that is applied to the end level data.
Level Multiplier	Scales the amplitude of the end level data.

Considerations

Before you use this window, you need to know some things about your playback file. It is a good idea to have a printed copy of your playback file. Consider the following:

- How many groups of end level data are there?
- Are the groups of data set up for multiple channels or multiple iterations of the same channel?
- Is the end level data in normalized units or engineering units?

Using the window

Perform the following to complete the file playback parameters window:

1. Press the File Name button and select a playback file (.sfp).
2. Enter the number of times you want to run the file in the Passes entry field.
3. Select a multiplier if the rate type value (time base) of the playback file is inappropriate for the test.
4. Select none if no compensation is needed.

   Select manual compensation if you want to manually adjust the mean level and full scale amplitude of the test. See the Manual Compensation run time window to use that window.

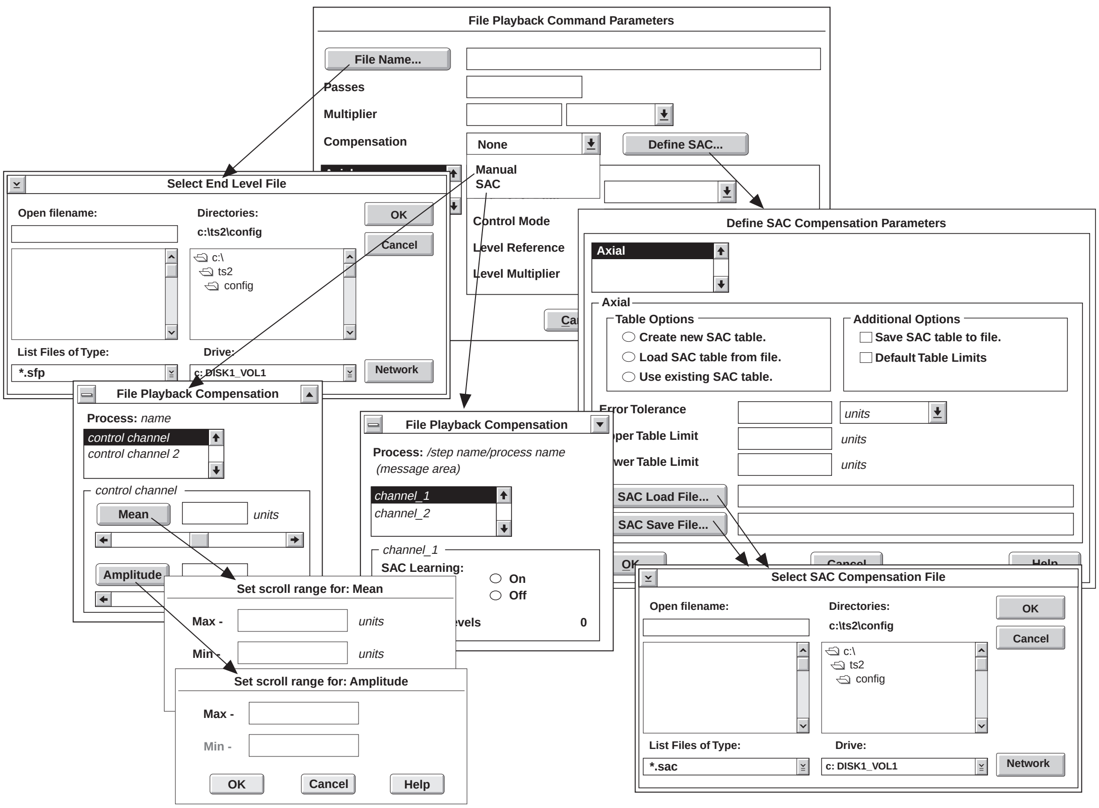
   Select SAC compensation if you want to use the spectrum amplitude control option. Press the Define SAC button to setup the SAC option. See the Define SAC Compensation Parameters window to use that window.
5. Select a control channel from the unlabeled scroll box.
6. Select the appropriate group of end level data.
7. Select the control mode for the channel.

Continued...

Using the window (continued)

8. If necessary, enter a level reference value to offset all of the end level data values.
9. If necessary, enter a level multiplier. Use a level multiplier when the end level data is entered as normalized units or the end level data is not optimized for the test.
10. If necessary, repeat steps 5 through 9 for each control channel.

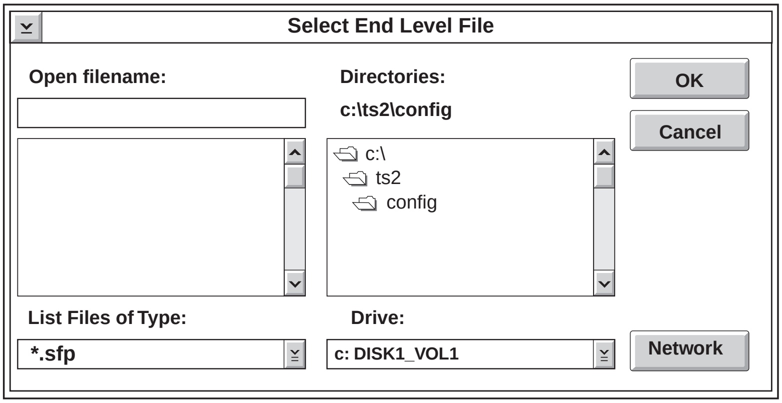
Subordinate windows



Select End Level File Window

Use this window to select the ASCII file for the file playback process.

Note:
The program you use to create the file playback file may add the extension .TXT to the file name. In this case, use either of the following to use a .TXT file.



- Rename the file to replace the .TXT extension with the .SFP extension.
- Replace the SFP extension in the File entry field of this window with the TXT extension. This causes all files with the .TXT extension to be shown in the Files list.

CONTROL	FUNCTION
Open filename	Displays ***.SFP** in the entry field. Select the file you want to open. All end level files use the extension **.SFP**. When you select a file, its name is shown in the entry field
files	Lists the end level files in the current directory. Selecting a file name displays it in the **File Name** entry field.
List Files of Type	Selects the type of files displayed in the File list. By default, ***.SFP** is selected. This displays only the files with the **.SFP** extension in the Files list.
Directories	Lists the available directories for the selected drive. Double-clicking a different directory displays the files of the directory in the **Files** list and any other directories in the **Directories** list.
Drive	Displays the current drive. All root directories of the drive are listed in the **Directories** list.
Network (Windows NT only)	Pressing the **Network** button displays the Connect Network Drive window where you can define new network drives and paths.

File Format

A playback file has special syntax requirements. See the syntax requirements for the header, channels, and data in the following pages. You can create a playback file using one of the following tools:

- a traditional text editor (except the OS/2 editor)
- a spreadsheet application
- an extraction from a data base

All of the above tools create ASCII files that contain data in a column-oriented syntax.

Note *You may want to add a .sfp extension to your playback files so they are easier to find.*

It is a good idea to have a printed copy of your file; for example, how many channels of data there are in the file. You should know the following when creating a playback file:

- Entries must be separated by a space or a tab.
- Blank lines between rows may be used to improve readability.
- Keywords (frequency, shape, etc.) can only be used once.
- Keywords are not case sensitive.
- There can be no space between a keyword used to define a constant and the equals sign (e.g, shape= sine). There can be a space after the equals sign.
- The TestStar configuration file should be compatible with the playback file (same number of channels, control mode units, etc.).
- The best file playback performance can be achieved when:
  - the waveshape is defined with a constant and no counters are used.

Or

  - fewer than 200 level data points define the profile (TestStar can read 200 points at a time before accessing the playback file again).

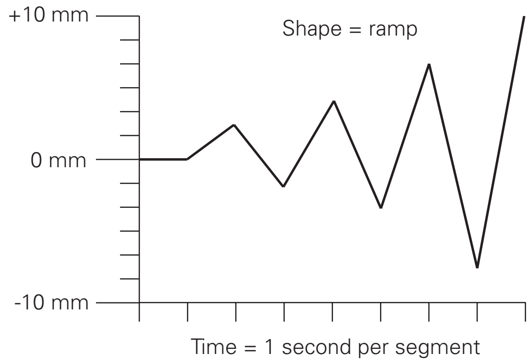
File format example

A playback file consists of a combination of constants and variables. Constants are not required.

This example shows two constants and one channel of data.

Since the shape and time are constants, only the level data needs to be a variable.

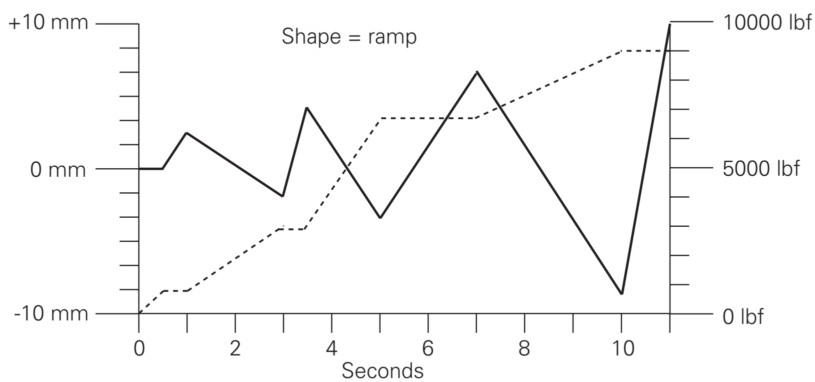
Constants	Shape= ramp Time= 1 sec
Variables	Level_Data1 mm
Data	0 3 -2 5 -4 8 -9 10



This example defines one constant and two channels of data.

The ramp is common to both channels. The time variable is also common for both channels.

Constant	Shape= ramp		
Variables	Time sec	Level_Data1 mm	Level_Data2 lbf
Data	0.5	0	0
	0.5	3	100
	2	-2	300
	0.5	5	300
	1.5	-4	700
	2	8	700
	3	-9	900
	1	10	900



Command data syntax

The command data follows the channel definition. Some of the keywords can be entered as constants or as individual commands. Constants are declared before any individual parameters are listed. Constants also require the equal sign (=) to be appended to the end of its keyword. Any keyword can be defined as a constant but only frequency, time, rate, and shape are practical constants. The following describe the keywords.

KEYWORD	RANGE	COMMENTS
Frequency **Time** **Rate***	undetermined	Required, only one of these three can be specified. *Rate must reflect the rate of change of the Level dimension.
Shape	Ramp, Haversine, or Step,	When not specified, default is sine, when not assigned as a constant, each segment can have a different shape.
Level_Data1	undetermined	Required
Level_Data2	undetermined	Optional
Level_Data3	undetermined	Optional
Level_Data4	undetermined	Optional
Counters	any name	Optional, displays a window showing all counters currently running.

Rate Type

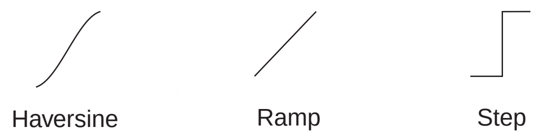
A rate type determines the time base of the waveform. Three types of rate expressions (keywords) are supported.

- FREQUENCY (Hz, cps)
- TIME (msec, sec, min)
- RATE (units per time; e.g.; kips/sec)

For example, a constant rate type of TIME= 2.5 sec assigns a time base of 2.5 seconds to each row of segment data. When a constant is defined, the related column (Frequency/Rate/Time) is not shown. A variable rate type identifies a column of time values, where each segment can have a different time base.

Shape

A waveshape defines how to go from the current end-level to the next end-level. The keyword SHAPE can define a waveshape as a constant at the beginning of the file (SHAPE= RAMP), or it can define a column of data to assign a shape to each segment end-level. Three waveshapes are supported. The SHAPE keyword should be the first constant or variable.



Level data

Up to four groups of end level data can be included in the file. When using variable attributes, end level data requires one of the labels LEVEL_DATA1, LEVEL_DATA2, LEVEL_DATA3, or LEVEL_DATA4 at the top of each column. These groups can be used for a four-channel test, or four separate tests in a single-channel system.

Note *End level data can be assigned a constant attribute. For example, LEVEL_DATA2= 500 lbf.*

Below each column label is the unit designation for the data. Each column of data includes a list of engineering units or normalized units that represent each command segment.

For example, assume you have a playback file that includes a column of force end levels. When you select that column with the End Level Data parameter, the only available control mode is force. If the column of data uses normalized values (percentage or unity), all control modes for the control channel are available.

Counters

Each end level data entry can be named with a counter label. The keyword COUNTERS is available to add counter names to selected end level data.

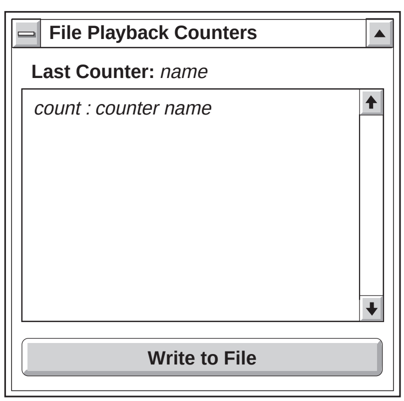
- A counter name is not case sensitive and can't have any spaces. The counter name is automatically capitalized when displayed.
- The single counter name can be used for multiple end levels within a playback file and/or in multiple files. Each time a counter name is encountered, its counter is incremented. Counter names are global.
- The COUNTERS keyword is the only keyword that allows blank entries in its column. COUNTERS can't be assigned a constant.
- Counter units must be specified as none.

Shape= haversine		
Time= 2.5 sec		
Level_Data1	Level_Data2	Counters
kN	mm	none
1000	4.5	
-1000	-4.5	flight1_takeoff
500	2.5	
-500	2.5	maneuver1
0	0	no load

Press the **Write to Data** File to output the counter data to the TestWare-SX data file.

Select **Zero Counters** in the Control menu to clear the counter value

Select **Reset** in the Control menu to clear the counter names.



When the counter keyword is specified, a run time window opens to show the file playback counter labels as they are encountered. The most recent counter that was played is shown in the Last Counter area. The seven digit count is listed first followed by the name of the counter

Variable attributes

A variable attribute is a column of attribute data with a keyword defining the column. Units for the attributes are in the row below the keywords. The order of the attribute columns is not important. *For example*, the attributes are placed in the first row with units assigned to each attribute in the second row.

Shape	Time	Level_Data1	Level_Data2
none	sec	lbf	lbf
ramp	.05	500	750
ramp	.05	-500	-750
haversine	.1	1000	1750
haversine	.1	-1000	-1750
step	1	0	0

Note *The string "none" is a place holder for the table. Any character string may be used.*

Constant attributes

Any keyword (except counters) can be defined as a constant. Append the equal sign (=) to the end of a keyword to define a constant. *For example*, the time attribute is a constant while the shape attribute is a variable. A playback file can contain constant and variable attributes, but constant attributes must precede variable attributes.

Time= 2.5 sec		
Shape	Level_Data1	Level_Data2
none	lbf	deg
ramp	1000	45
ramp	-1000	-45
haversine	500	22.5
haversine	-500	-22.5
step	0	0

Normalized units

A playback file can have normalized segment data. Normalized units are percentage (%) or unity (a multiplier value). A file containing normalized data uses the Level Multiplier to assign units and scale the waveform.

Note *Using the % symbol with strain represents the percentage of the strain units - not the percent strain.*

For example, assume you have a playback file that contains a column of data with percentage values and another column has unity values.

Shape= haversine
Time= 2.5 sec

Level_Data1	Level_Data2
%	unity
100	1
-100	-1
50	.5
-50	-.5
0	0

If you set the Level Multiplier to 1000 lbf and select either column (End Level Data in the Parameters window), the waveform would cycle between ±1000 lbf, then between ±500 lbf.

Note *You could assign the end level data as a constant, but this is the same as defining a monotonic command. The following would ramp from 0 to 1000 lbf in 2.5 seconds and hold at 1000 lbf.*

Shape= ramp
Time= 2.5 sec
Level_Data1= 1000 lbf

File Playback Compensation Window (manual)

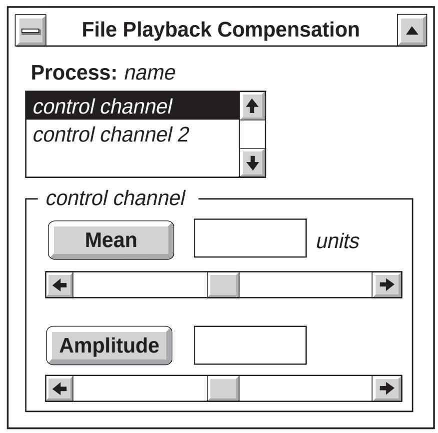
Prerequisite

The compensation selection in the File Playback Command window must be set for Manual.

Note *This is the run-time window for manual file compensation. Another run-time window with the same name is used for SAC compensation.*

A run-time window opens for each file playback process called in a test. All the windows are opened before the test begins.

Use this window to manually adjust the mean level offset and amplitude of the file playback process.



Using the window

This window appears when a file playback process is being executed and the compensation function is enabled.

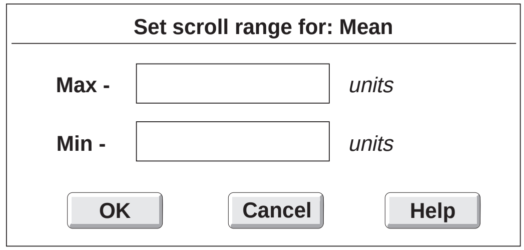
- Select a control channel, then adjust the controls for that channel. One channel can be adjusted at a time.
- Pressing the Mean button displays a Set Scroll Range window where you can change the maximum and minimum values of the Mean adjustment.
- Pressing the Amplitude button displays a Set Scroll Range window where you can change the maximum value of the Amplitude adjustment.

Set Scroll Range Window

The same basic window is used to scale the Mean and Amplitude controls.

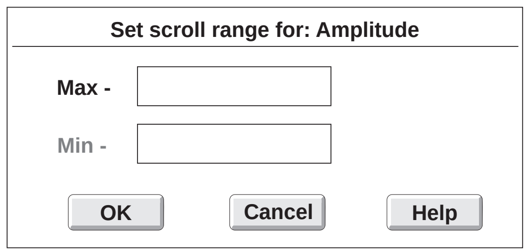
This window changes the range of the Mean adjustment.

Set the range to ensure that the mean level control can't be adjusted beyond a safe range.



This window changes the range of the Amplitude adjustment.

The amplitude adjustment is a simple multiplier. Only the Max adjustment is used.



CONTROL	FUNCTION
Max	Sets the highest value of the Mean control. The maximum setting affects the right side of the Mean scroll bar. Sets the multiplier of the Amplitude control. The maximum setting affects the right side of the Amplitude scroll bar.
Min	Sets the lowest value of the Mean control. The minimum setting affects the left side of the Mean scroll bar. This is not used for the Amplitude control.
units	Displays the units of the selected control mode. Amplitude does not use units.

Using the window

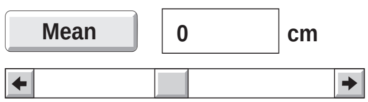
Note *This example describes how to scale the Mean control. Scaling the Amplitude control operates in a similar way. The Amplitude control is scaled by a multiplier, the standard amplitude scaling is x2. Use the set scroll range window to change the maximum multiplier.*

The window displays the full-scale value of the selected control mode range. You can change the maximum and/or minimum setting of the Mean control to a value less than full-scale. The Max and Min limits can be set to different levels.

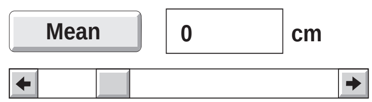
You cannot set the maximum range to a value less than the current Mean setting or the minimum range to a value more than the current Mean setting. Reducing the scroll range also increases the resolution of the control adjustment.

To change a scroll range value, type the new value in the entry field.

For example, assume the control mode range is ±3 cm.



With the Max and Min values set to +3 cm and -3 cm, the zero reference is centered.



With the Max and Min values set to +3 cm and -1.5 cm, the zero reference is offset.

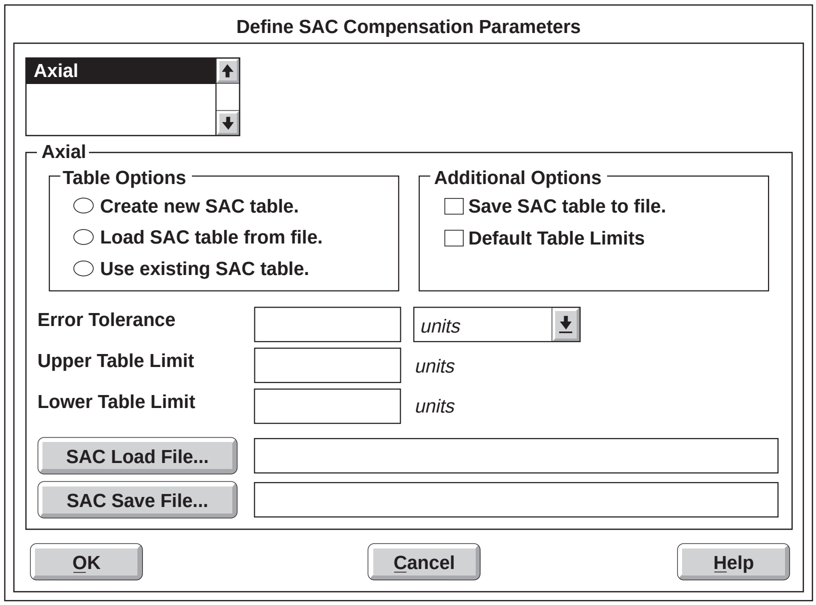
Define SAC Compensation Parameters Window

Prerequisite

The compensation selection in the File Playback Command Parameters window must be set for SAC. The Model 490.50 Processor module must have the T805 processor module (revision B or C of the module).

The purpose for the SAC (spectrum amplitude control) option is to better meet the end levels in a spectrum/random waveform.

Use this window to create, save, select and use a SAC table.



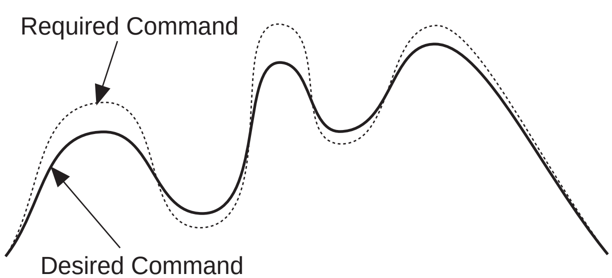
CONTROL	FUNCTION
control channel	Displays a list of the available control channels. The SAC file can be applied to one control channel. Selecting a control channel displays its name for the compensation parameters of the process.
Table Options	Specifies what you wish to do with a SAC table file.
Additional Options	Enabling **Save SAC table to file** writes the current SAC file to disk when the process ends. This option is available for all three table options. Enabling **Default Table Limits** sets the upper and lower table limits to the range of the control mode feedback signal. This option is available only when you are creating a an SAC table.
Error Tolerance	Specifies a range for the missed end levels count in the File Playback Compensation window.
Upper and Lower Table Limits	Shows the limits of the SAC table when it was created. The default setting is 0 which is 100%.

CONTROL	FUNCTION
SAC Load File	Displays the Select SAC file window where you open an SAC table file. The complete file path is displayed in the entry field below the button.
SAC Save File	Displays the Select SAC file window where you save the SAC table file. The complete file path is displayed in the entry field below the button.

How SAC works

Spectrum Amplitude Control (SAC) is an algorithm that uses a table while the test is running to keep track of end levels. As the test is running, the controller monitors the test progress and compares each command end level with its feedback end level. The difference between the command end level and its feedback is continuously monitored and the over-programmed level is adjusted continuously.

An SAC compensation table contains an over programming value required to reach each desired end level



Note *It takes about 5 seconds to load or save a SAC compensation file for each control channel. This delay occurs whenever a file playback process begins or ends.*

The error tolerance sets the parameters for the missed end levels counter. Whenever the feedback end level is outside the error tolerance band, the counter is incremented.

Table options

The Additional options can be used in conjunction with the Create, Load, and Use options.

Create

Use this option (along with the Save SAC table to file option) the first time you run the playback file.

Load

Use this option to load a SAC table you previously created for the playback file. Loading an SAC table also displays the upper and lower limits that were assigned when the table was created.

Use

Use this option when you want to use the SAC table that currently exists in memory when this process begins. This is useful when one SAC table has been optimized for a test with multiple playback files. It also displays the upper and lower limits that were assigned when the table was created.

Note *When you have the* ***Save SAC table to file*** *enabled, you create a new SAC table when the process is complete.*

Table limits

When the Default Table Limits option is enabled, the SAC table is created for the selected range of the input sensor used for the control mode. When the Default Table Limits option is disabled, you enter the upper and lower limits of the SAC table. This is useful when you know the range of your test.

For example, assume you are using a force sensor with a selected range of ±20 kN. The default table limits are +20 kN and -20 kN. If your test runs within +1 kN and +12 kN you will only be using a portion of the table. Creating an SAC table with limits of +1 kN and +12 kN optimizes the table for the test.

If you know the range of your test, disable the Default Table Limits option and enter your limits in the appropriate entry fields. This creates an SAC table with more data points that apply to your test as opposed to a table with the same number of data points applied to the range of the sensor.

⚠ WARNING

TestStar DOES NOT apply any overprogramming to end levels outside the SAC table limits.

This applies when the Default Table Limits option is disabled. Be sure the limits you enter exceed the end levels specified in the data file.

File Playback Compensation Window (SAC)

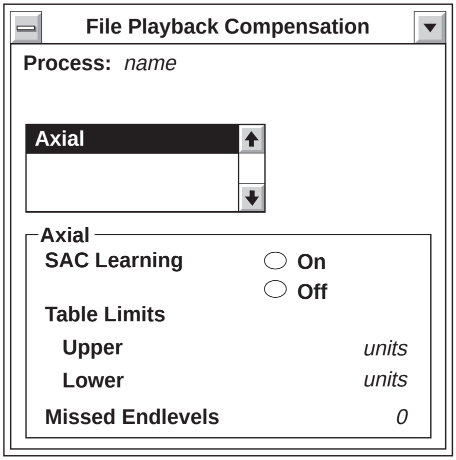
Prerequisite

The compensation selection in the File Playback Command Parameters window must be set for SAC.

Note *This is the run time window for SAC file compensation. Another run time window with the same name is used for manual compensation.*

A run time window is opened for each file playback process called in a test. All the windows are opened before the test begins.

Use this run-time window to select the SAC Learning function and monitor missed end levels.



CONTROL	FUNCTION
Process	Displays the name of the step and the name of the file playback process. Below this is a message area to indicate when a SAC file is being loaded or saved.
control channels	Displays the control channel that the SAC table is being applied to.
SAC Learning	Enables or disables the SAC Learning function.
Table Limits	Shows the upper and lower limits associated with the current SAC table.
Missed End Levels	Shows the number of end levels that were not within the error tolerance range. This information is added to the data file.

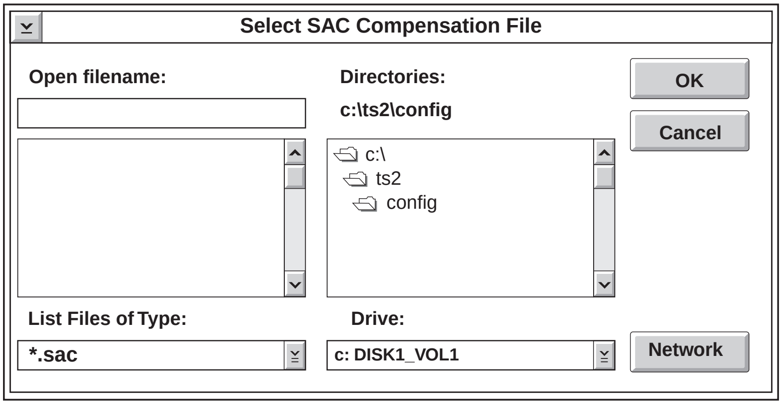
SAC Learning

The learning function optimizes a SAC table after the table has been created. The learn function compares each playback end level plus its associated SAC value with the actual end level and calculates a new SAC level. If the save option is selected in the SAC Compensation Parameters window, the revised SAC table replaces the current SAC file.

If SAC Learning is off, the algorithm does not update the current SAC table.

Select SAC File Window

Use this window to load and save a spectrum amplitude control file for the file playback process.



Control	Function
Open filename	Displays ***.SAC** in the entry field. Type the name you want to call the template. All template files use the extension **.SAC**. When you select a file, its name is shown in the entry field
files	Lists the template files in the current directory. Selecting a file name displays it in the **File Name** entry field.
List Files of Type	Selects the type of files displayed in the File list. By default, ***.000** is selected. This displays only the files with the **.000** extension in the Files list.
Directories	Lists the available directories for the selected drive. Double-clicking a different directory displays the files of the directory in the **Files** list and any other directories in the **Directories** list.
Drive	Displays the current drive. All root directories of the drive are listed in the **Directories** list.
Network (Windows NT only)	Pressing the **Network** button displays the Connect Network Drive window where you can define new network drives and paths.

Hold Command

A hold process produces a command signal that controls a servovalve or servo motor. This process should be sequenced in series with other command processes.

Note *Pages 230 through 233 provide details common to all processes such as default settings, using triggers, and how to reach the process windows.*

Contents

How it works

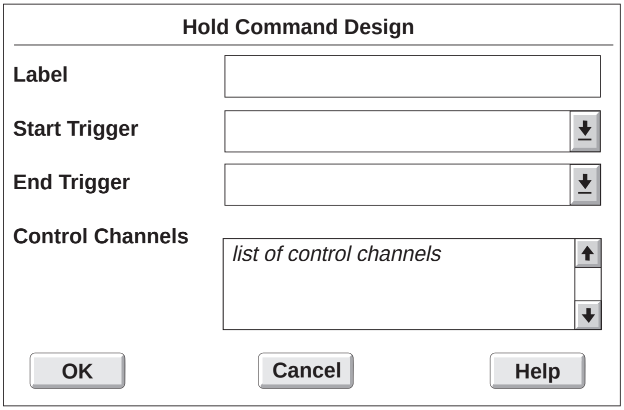
The hold command maintains a static level for a specified amount of time. When a hold command begins, it reads the current level of the selected control mode and maintains that level.

Hold Command Design Window

This window specifies when the process starts and stops. The information in this window is saved with the test template.

Note *Starting two test commands at the same time for the same control channel will display a message box that describes an error condition.*

Use this window to create a process to hold the test command at a level for a time.



CONTROL	FUNCTION
Label	Names the process. Type the name you want to call the process in the entry field.
Start Trigger	Specifies the beginning of the process. Press the list icon and select a trigger.
End Trigger	Specifies the end of the process. Press the list icon and select a trigger.
Control Channels	Specifies the control channels that the process is to be applied (more than one may be selected). A list of the available control channels is shown.

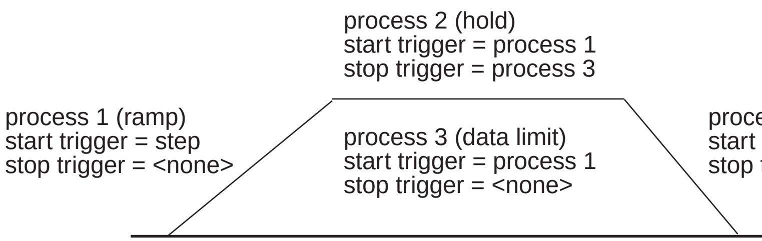
Trigger example

Suppose you created four processes with the following attributes (only the attributes relevant to the example are listed):

PROCESS 1	PROCESS 2	PROCESS 3	PROCESS 4
ramp to +2 cm	hold for 10 seconds	trigger if output drops 0.5 N	ramp to 0 cm
length control	length control	monitor force	length control

Process 1 ramps to +2 cm. Processes 2 and 3 start together. Process 2 holds at 2 cm while process 3 monitors a change in force.

If process 3 detects the required change, it triggers the end of process 2. If process 3 does not detect a change, process 2 times out and ends (which also stops process 3). When process 2 ends, process 4 ramps back to zero.

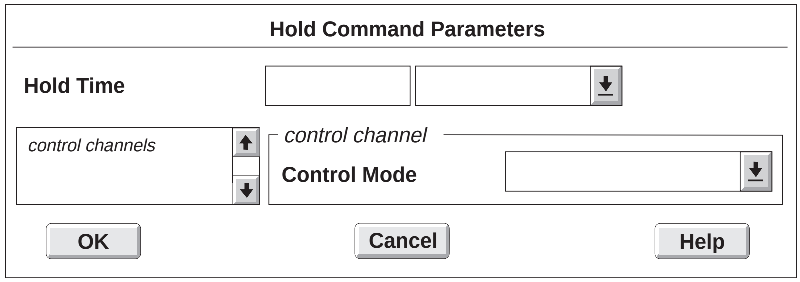


Hold Command Parameters Window

Prerequisite

You must create a hold command process in a template with the design window before you can access this window. The information in this window is saved with the test procedure.

Use this window to define the parameters of the process.



CONTROL	FUNCTION
Hold Time	Specifies the time for the hold period. Type the value of the hold time in the entry field. Pressing the list icon shows the available selections for the time units.
control channels	Displays a list of the available control channels. The hold command can be applied to more than one control channel. A control mode can be selected for each control channel.
Control Mode	Specifies the control mode for the process. Pressing the list icon shows the control modes defined with the Controller Definition function.

Using the window

Use the right entry field to select the desired units. Enter the hold time in the left entry field. Select a control channel and a control mode entry for each control channel.

Monotonic Command

A monotonic command process produces a command signal that controls a servovalve or servo motor. This process should be sequenced in series with other command processes.

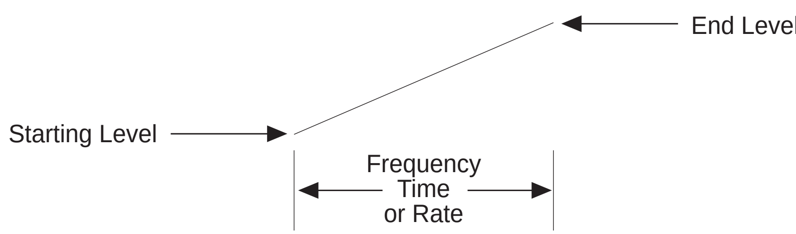
Note *Pages 230 through 233 provide details common to all processes such as default settings, using triggers, and how to reach the process windows.*

Contents

How it works

A monotonic command starts at the current level and ends at another level. You assign one of three wave shapes, a time, and a control mode to a monotonic command.

The starting level is usually the end level of the previous command process.

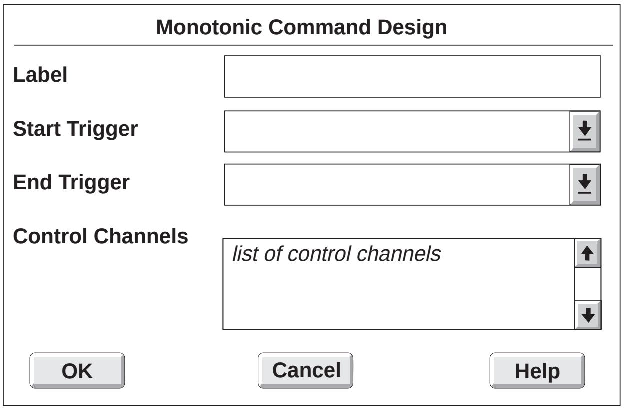


Monotonic Command Design Window

This window specifies when the process starts and stops. The information in this window is saved with the test template.

Note *Starting two test commands at the same time for the same control channel will display a message box that describes an error condition.*

Use this window to designate a single test command process in a step.



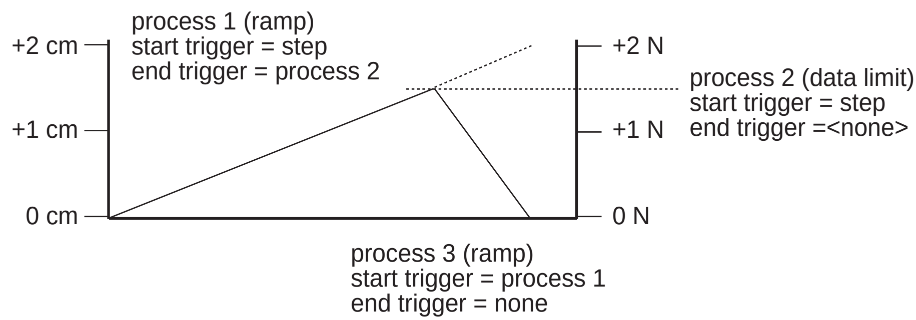
CONTROL	FUNCTION
Label	Names the process. Type the name you want to call the process in the entry field.
Start Trigger	Specifies the beginning of the process. Press the list icon and select a trigger.
End Trigger	Specifies the end of the process. Press the list icon and select a trigger.
Control Channels	Specifies the control channels that the process is to be applied to (more than one may be selected). A list of the available control channels is shown.

Trigger example

Assume you created three processes with the following attributes (only the attributes relevant to the example are listed):

PROCESS 1	PROCESS 2	PROCESS 3
ramp to +2 cm	trigger if greater than +1.5 N	ramp to zero
length control	monitor force	length control

Process 1 ramps to +2 cm if process 2 does not detect its limit. If process 2 detects a force greater than +1.5 N then processes 1 and 2 end. When process 1 ends, process 3 begins.

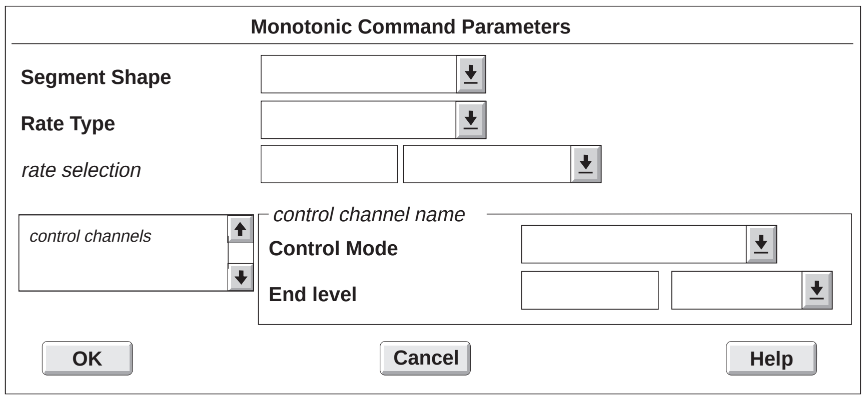


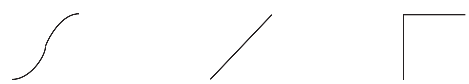
Monotonic Command Parameters Window

Prerequisite

You must create a monotonic command process in a template with the design window before you can access this window. The information in this window is saved with the test procedure.

Use this window to define the parameters of a single test command process.



CONTROL	FUNCTION
Segment Shape	Specifies a single test command that starts at one level and ends at a different level. The relative segment shapes let you define the command amplitude with relative end levels.  Haversine Ramp Step
Rate Type	Specifies what type of units you can use to define a segment. The types of expressions are: • Frequency • Time • Rate
rate selection **Frequency** **Time** **Rate**	Specifies the monotonic command value. The label for this field is the name of the selected Rate Type. Press the list icon of the right entry field and select the units you want for the rate type value. Type the value you want to specify for the command in the left entry field.
control channels	Displays a list of the available control channels. The monotonic command can be applied to more than one control channel. The control mode and end level can be selected for each control channel.

CONTROL	FUNCTION
Control Mode	Specifies the control mode for the process. Pressing the list icon shows the control modes you defined with the Edit Control Modes window.
End level	Specifies the end level of the segment. The starting level is the ending level of the previous test command. An end level can be specified with an absolute value or relative value (depending on the segment shape selection).

Time and frequency

The rate type of time specifies the time to execute one segment. The rate type of frequency specifies the time to execute a two-segment cycle.

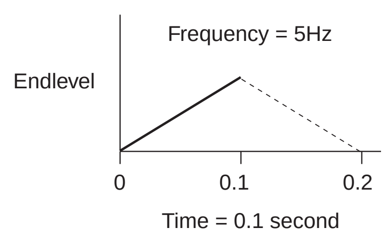
Time Units
milliseconds
seconds
hours
days

Frequency Units
hertz (Hz)
cycles per second (cps)
cycles per minute (cpm)
cycles per day

Selecting time or frequency changes the rate type value label to Time or Frequency respectively. Select the units of measurement to the right of the rate type value field. Both rate types shown specify the same waveform using different types of values.

The units for **Frequency** are based on two segments (although only one segment is produced)

The units for **Time** are based on one segment.

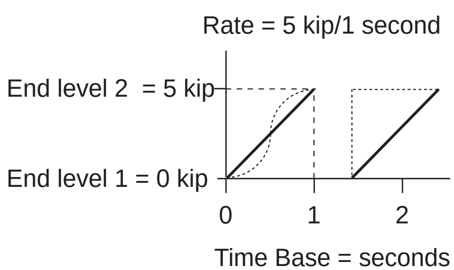


Rate

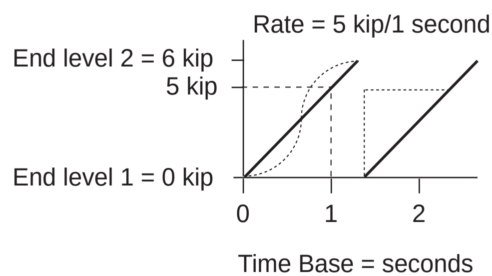
Selecting rate allows you to specify a segment with a constant rate between the starting level and the end level. A rate value represents the amount the control mode changes in one time unit.

Rate is typically associated with a ramp. You specify the rate as if a ramp were being programmed. A haversine or step (square wave) segment shape is maintained when you use rate.

Selecting rate changes the rate type value label to reflect the selected control mode. Select the units of measurement to the right of the rate type value field.



A rate is expressed as units per time. (A ramp is shown along with a haversine and step segment.)



The rate is maintained even when the end level has changed

End level

Assume a monotonic command begins with displacement at +2 inches. If you define a ramp with an end level of - 2 inches, the command ramps from +2 - -2 inches. If you define a *relative* ramp with an end level of - 2 inches, the command ramps from +2 - 0 inches.

Operator Event

The operator event process is an event process. It can be sequenced in series like a command process or in parallel like other processes.

Note *Pages 230 through 233 provide details common to all processes such as default settings, using triggers, and how to reach the process windows.*

Contents

How it works

An operator event process produces a window that appears when the test begins (a run time window). This window displays up to three buttons that you can use to affect the flow of a test. The operator event process provides a way for you to manually jump to a different part of a test.

The information in the Operator Event window is also displayed on the load unit control panel. The three buttons represent three separate operator events. A test can have any number of operator events providing that no more than three events are active at one time.

Here are some of the things you can do with an operator event process:

- prompt the operator before the test begins
- trigger the beginning of a data acquisition process
- manually trigger the end of a command process
- manually set a digital output channel
- hold a test indefinitely

A single shot process ends when you press the button once. This grays out the button and removes the description. Once the operator event ends, another operator event can use the disabled button.

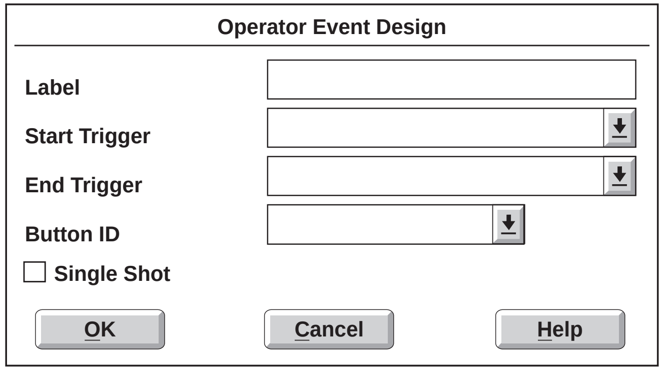
If the operator event is not configured as a single shot process, it continues until encountering an end trigger or until the step ends.

When you run your template, a check is made to be sure that no two operator events using the same button at the same time. The start trigger of one process cannot begin until the end trigger of the other process is complete.

Operator Event Design Window

This window names the process and specifies how the process starts and stops.

The information in this window is saved with the test template.



CONTROL	FUNCTION
Label	Names the process. Type the name you want to call the process in the entry field.
Start Trigger	Specifies the beginning of the process. Press the list icon and select a trigger.
End Trigger	Specifies the end of the process. Press the list icon and select a trigger.
Button ID	Specifies which of the three buttons is linked to this process.
Single Shot	If this is enabled, the button can be pressed once and the process ends. Otherwise the button can be pressed any number of times and the process ends when the step ends.

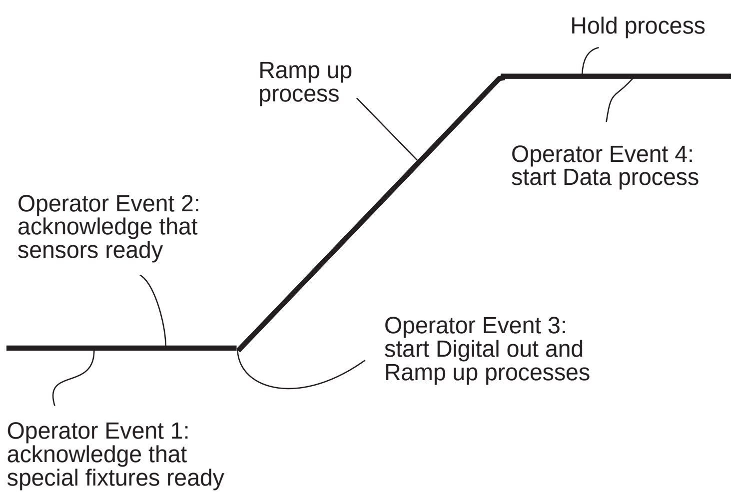
Trigger example

The test begins by displaying an operator event dialog box with three functions. Review how the triggers work for the event processes.

Processes	Event 1	Event 2	Event 3	Event 4
Start Trigger	Step Start	Event 1	Event 2	Event 3
End Trigger	Event 3	Event 3	<none>	<none>
Button	1	2	3	1

Processes	Digital out	Ramp up	Hold	Data
Start Trigger	Event 3	Event 3	Ramp up	Event 4
End Trigger	<none>	<none>	<none>	<none>

Assume you have a simple test that has the characteristics shown.



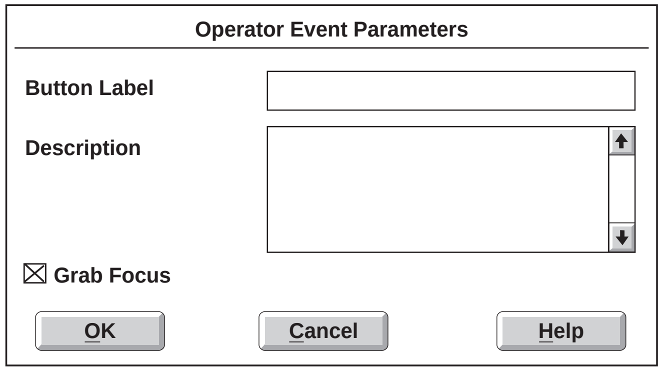
- Event 1 prompts the operator to check a condition and enables event 2 when the button is pressed.
- Event 2 prompts the operator to perform a task and enables event 3 when the button is pressed.
- Event 3 starts an external recorder, starts the ramp command, and enables operator event 4.
- Event 4 allows you to start a data acquisition process any time during the ramp or hold processes.

The Processes

Operator Event Parameters Window

You must create an operator event process in a template with the design window before you can access this window. The information in this window is saved with the test procedure.

Use this window to assign a name to the button and add text to describe the process function.



CONTROL	FUNCTION
Button Label	Names the button. Type the name you want to call the button in the entry field. A button label must be 9 characters or less.
Description	Describes the function of the button. The description is located next to the button in the dialog box. A description can be up to 256 characters although only about 80 characters are visible.
Grab Focus	The runtime window becomes active (to the front) when the process begins. This is selected by default.

Using the window

Type a name in the button label entry field. Enter a description that describes the use of the button. Each operator event is a separate process.

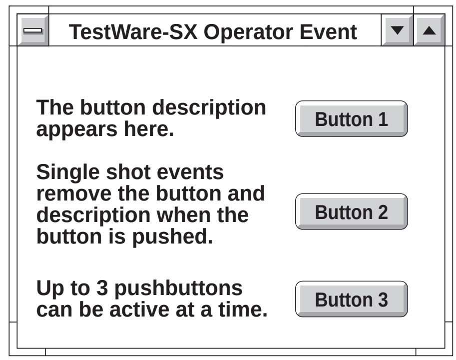
Enable the Grab Focus checkbox to ensure that the dialog box is visible when the operator event process starts.

Run-time window

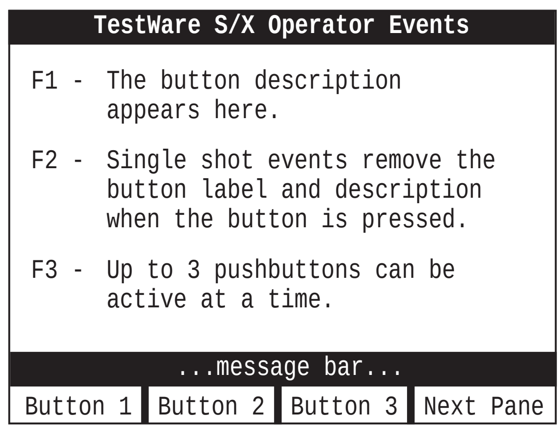
When you run the test procedure, the TestWare-SX Operator Event window opens. Pressing an event button causes the associated process to end. When an event process ends the associated text disappears and the button is "grayed out." This allows another process to use the button.

This window appears when the test starts.

This window shows three separate operator events.



The operator event process also appears in the load unit control panel display.



Operator Information

The operator information process is a special process. It can be sequenced in series like a command process or in parallel like other processes.

Note *Pages 230 through 233 provide details common to all processes such as default settings, using triggers, and how to reach the process windows.*

Contents

The Processes

How it works

The operator information process produces a window that appears when the process is called during a test procedure (a run-time window). The window contains several lines of information (this can also be called a form). Some of the lines of information are non-editable while other require the operator to enter specific information or comments.

When the operator presses the OK button, the information is output to the test data file.

Here are some of the things you can do with an operator information process:

- Prompt the operator for information, then begin the test. Or prompt the operator for information at the end of a test.
- Provide specimen information with the data file.
- Prompt the operator for specific information such as:
  - an operator ID
  - a batch number
  - a workstation number
  - part number
  - operator comments
- Provide QC information with the test data.

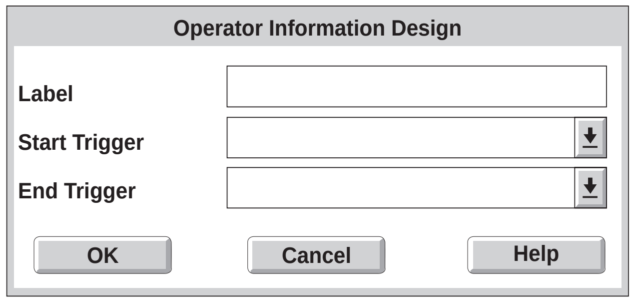
While the operator information process can be used to trigger other processes, its primarily purpose is to provide specific information about a test with the test data. While this process is similar to the Description selection in the Data menu, it allows you to create a standard format (or form) for the information.

Operator Information Design Window

This window names the process, specifies how the process starts and how the process stops.

Note *Only one Operator Information process can be active at a time; otherwise, an error is generated.*

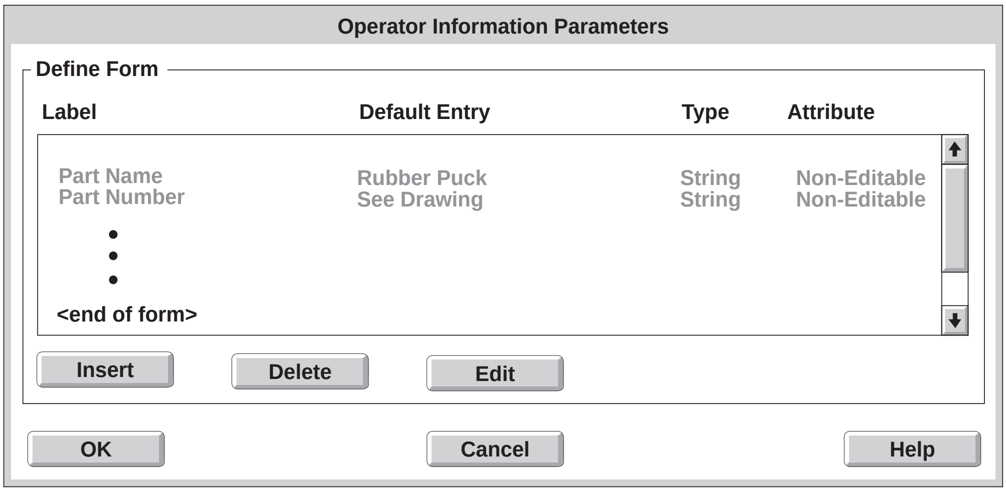
The information in this window is saved with the test template.



CONTROL	FUNCTION
Label	Names the process. Type the name you want to call the process in the entry field.
Start Trigger	Specifies the beginning of the process. Press the list icon and select a trigger.
End Trigger	Specifies the end of the process. Press the list icon and select a trigger.

Operator Information Parameters Window

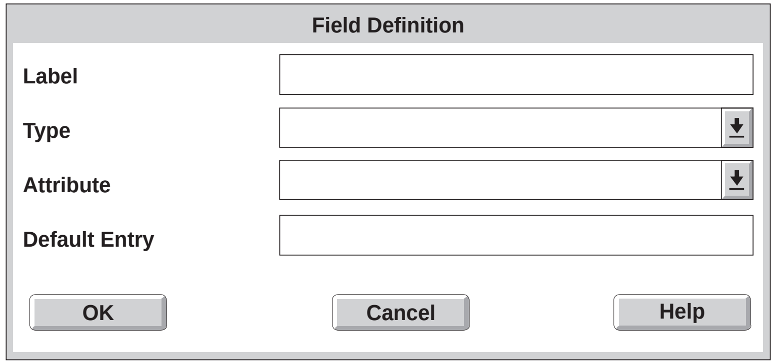
You must create an operator information process in a template with the design window before you can access this window. The information in this window is saved with the test procedure. Use this window to create a run-time window that requires the operator to enter specific information.



CONTROL	FUNCTION
Define Form	The four headings in this area are defined with the Field Definition window. The list of items in the form appear (in the order listed) in the run-time window.
Insert	Opens a new Field Definition window where you to insert a new form field immediately in front of the selected field.
Delete	Deletes the selected field. Be *sure* you no longer need the field before deleting it: it cannot be recovered!
Edit	Opens the existing Field Definition window for the selected field. You can also open this window by double-clicking on the desired field.

Field Definition Window

Use this window to define each line of the Operator Information run-time window.



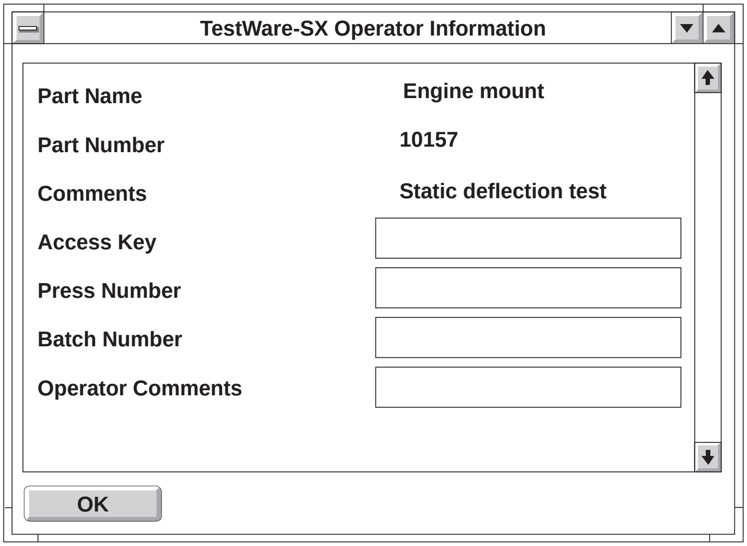
CONTROL	FUNCTION
Label	This is the text that describes the field. It can contain up to 48 characters.
Type	This defines the type of information in the field. The only selection is *String*.
Attribute	This defines how the field can select the operator input. Choices are: ***Non-Editable***—text is for display only, and cannot be changed by the operator. ***Non-Blank***—operator can enter any information, but cannot leave the field empty. ***None***—Operator can edit or delete the default entry.
Default Entry	This is the initial entry displayed for the operator. It can contain up to 48 characters.

Operator Information Window

The information shown in this window is defined with the Parameters window and the Field Definition window

Some items provide non-editable information about the test while other items require the operator to enter specific kinds of information.

This run-time window is displayed whenever the operator should enter information during a test.



Using the run-time window

This window automatically opens in front of the TestWare-SX Execute Procedure window when the test requires an input from the operator. Enter any required information and press OK. The following actions occur:

- The data on the form is transferred to the data file
- The run-time window closes.

Peak/Valley Change Detector

The peak/valley change process is a detector process and should be sequenced in parallel with command processes.

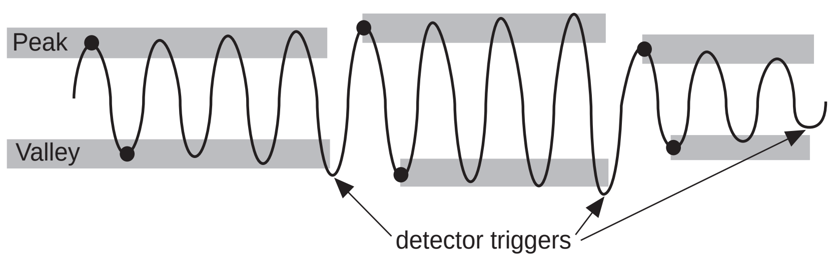
Note *Pages 230 through 233 provide details common to all processes such as default settings, using triggers, and how to reach the process windows.*

Contents

How it works

This process lets you monitor an input signal for changes in the peaks and valleys. The process begins by detecting a peak and valley, these values become the reference levels for the tolerance range. When a peak or valley exceeds a tolerance range, the process triggers. If the process is set for continuous triggers, it detects new reference levels for the tolerance setting and continues the process.

The detector tolerance is shown in gray. Peak and valley reference values are shown as dots.

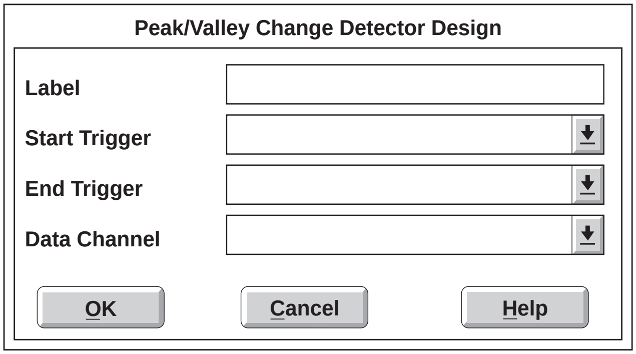


This process is useful during cycle commands to monitor input signals for specific changes that trigger other processes. The process can automatically set new tolerance reference values as the input signal changes when using the trigger continuously option.

Peak/Valley Change Detector Design Window

This window names the process and specifies how the process starts and stops.

The information in this window is saved with the test template.



Control	Function
Label	Names the process. Type the name you want to call the process in the entry field.
Start Trigger	Specifies the beginning of the process. Press the list icon and select a trigger.
End Trigger	Specifies the end of the process. Press the list icon and select a trigger.
Data Channel	Selects an input signal to monitor. Do not select the input signal being used for the control mode.

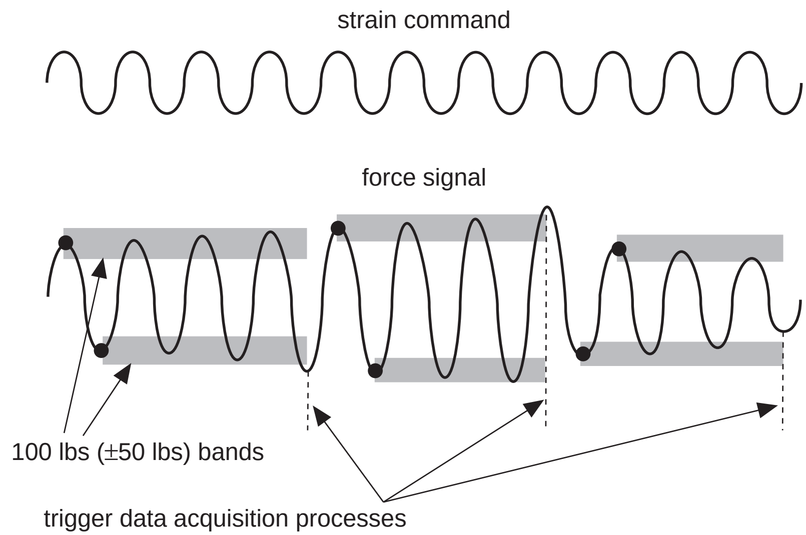
Trigger example

Suppose you want to acquire a cycle of data each time the peaks or valleys of an input change a specified amount. You can create a peak/valley change detector with the following parameters:

- Trigger continuously
- Tolerance of ±50 lbs.
- Peak sensitivity of 5 lbs.

For example, the test command cycles between ±0.0001 strain and you are monitoring a force channel

Each time the force signal changes more than ±50 lbs. a trigger is issued to a data acquisition process.

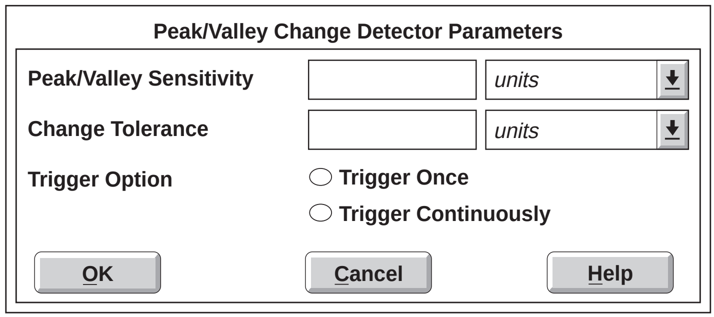


The **start trigger** would be step start or the same start trigger as the strain command process. The **end trigger** would be none. The data acquisition process would use the peak/valley change detector process as its start trigger. A single data acquisition process could be run multiple times with this change detector.

Peak/Valley Change Detector Parameters Window

You must create a peak/valley change detector process in a template with the design window before you can access this window. The information in this window is saved with the test procedure.

Use this window to establish the detector characteristics.

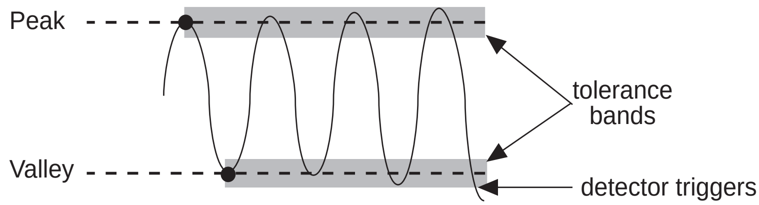


CONTROL	FUNCTION
Peak/Valley Sensitivity	Determines when a peak or valley is detected. The sensitivity setting is the amount the signal must change to detect a peak or valley. Select the units you want for the sensitivity value.
Change Tolerance	Specifies the tolerance band around the reference peak and valley values. The tolerance value represents a plus and minus deviation from the reference value.
Trigger Option	Trigger Once ends the process when the first peak or valley exceeds the tolerance setting. Trigger Continuously issues a trigger each time a peak or valley exceeds the tolerance setting. This selection ends according to the end trigger selection.

Tolerance

The tolerance value is applied to the peak reference level and the valley reference level.

The first peak and valley detected by the process become the reference levels for the tolerance band.



Using the window

- Enter a sensitivity value. Setting the Sensitivity too high may cause low-amplitude signal changes to be missed. Setting the Sensitivity too low may cause signal noise to be recognized as a peak/valley value.
- Enter a tolerance value. This value determines the requirements to issue a trigger.
- Select a trigger option. The trigger option lets you issue one or more triggers.

Program Control

The program control process is a special process and should be sequenced in parallel with command processes.

Note *Pages 230 through 233 provide details common to all processes such as default settings, using triggers, and how to reach the process windows.*

Contents

How it works

The program control process works like a custom interlock. Use this process to stop a test before it is complete. It is commonly used in conjunction with other event detector processes.

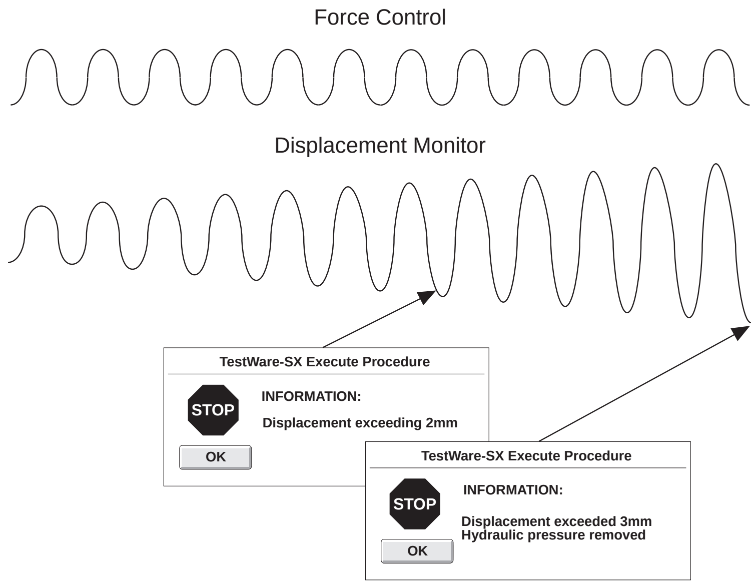
Note *The other TestStar processes can prematurely end a step but not the test program. This process offers several levels of program control.*

For example, assume you have a test that consists of several steps. You can configure this process to end the test during one of the early steps if an undesirable condition is detected.

While the test is operating in force control, two limit detector processes monitor the displacement sensor and trigger a program control process.

One program control process displays a message when the sensor output exceeds 2 mm.

The second program control process causes a hydraulic interlock when the sensor output exceeds 3 mm.

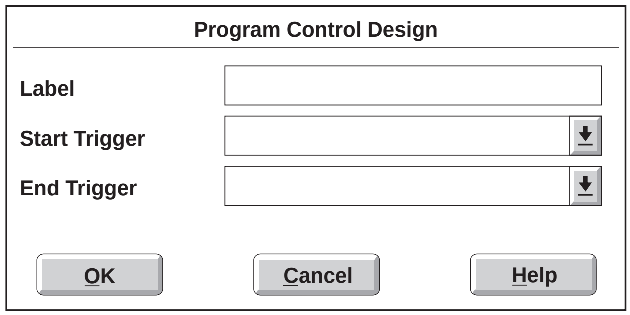


Program Control Design Window

This window names the process and specifies how the process starts and stops.

Note *The start trigger for this process is usually the end trigger of an event process or data acquisition process.*

The information in this window is saved with the test template.



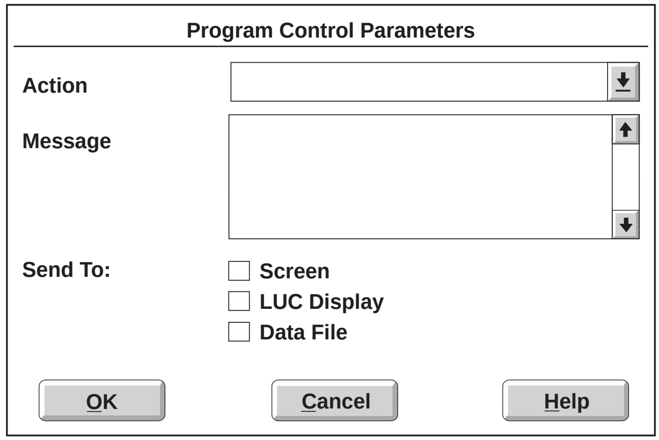
CONTROL	FUNCTION
Label	Names the process. Type the name you want to call the process in the entry field.
Start Trigger	Specifies the beginning of the process. Press the list icon and select a trigger.
End trigger	Specifies the end of the process. This process performs its action immediately and ends, therefore use the none trigger.

Program Control Parameters Window

You must create a program control process in a template with the design window before you can access this window. The information in this window is saved with the test procedure.

Use this window to select an action that will happen when the process executes.

Select an action, enter a message, and select where the message is displayed.



CONTROL	FUNCTION
Action	Selects the action that the process performs. Pressing the list icon shows the selections.
Message	Enters the message that is displayed (or recorded) when the process occurs.
Send To	Selects where the message occurs. Any combination of the three locations can be selected. **Screen**Displays the message in a dialog box on the computer screen. **LUC Display**Displays the message in the message bar on the load unit control panel. **Data File**Writes the message to the data file. This is ignored if a data file is not used.

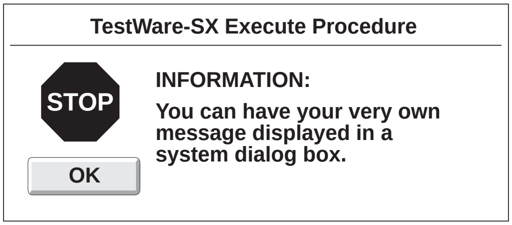
Actions

The program control process has 6 levels of control.

Action	Description
None	Disables the process.
Message Only	Displays a message window.
Program Hold	Suspends the test. It functions the same as pressing the Hold button.
Program Stop	Stops the test. It functions the same as pressing the Stop button.
HPS Off	Stops the test and removes hydraulic pressure at the pump.
Error	Stops the test and generates an error. TestWare-SX must be reset using the Control menu.

All program control processes (except none) display a dialog box with the message specified in the parameters window.

Simply press OK to remove the dialog box.



Step Done Definition Window

A Step Done Definition process is required for each step. You must specify at least one process to cause the step to finish.

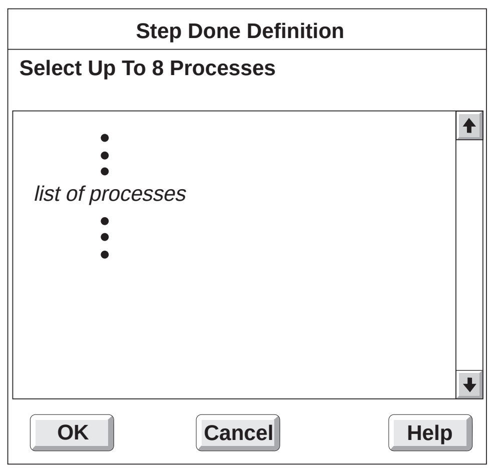
Any process in the step can be selected to jump to the Step Done process. Up to eight processes may be selected to end the step. If any process selected in the Step Done Definition window reaches its end trigger, the step ends.

Prerequisite

You must design a process before you can use this window. The Step Done Definition window can only be reached when working with a template.

You must design a process before you can use this window. The Step Done Definition window can only be reached when working with a template.

Use this window to specify which processes can end the step.



Using the window

At least one process must be selected to end the step. Normally, you would select the last test command process as the trigger for the Step Done process.

You may select up to eight processes to end the step. If any of the selected processes ends, the Step Done process is encountered and all processes are terminated. Typically, data processes are selected to prematurely end the step.

Temperature Control

The temperature control process is an external control process. Use it to set the temperature in an environmental chamber or a furnace.

Note *Pages 230 through 233 provide details common to all processes such as default settings, using triggers, and how to reach the process windows.*

Contents

Prerequisite

This process is designed to be used with an MTS Model 409.80 or 409.82 Temperature Controller with the RS-232 interface option (p/n 475319-01). A serial port must be specified with the hardware configuration mode of the TestStar software installation program.

To use the monitor tolerance you must have an input signal configured as an external input (see Chapter 10 in the TestStar Installation manual) and an external conditioner that processes a temperature sensor into a linear analog voltage.

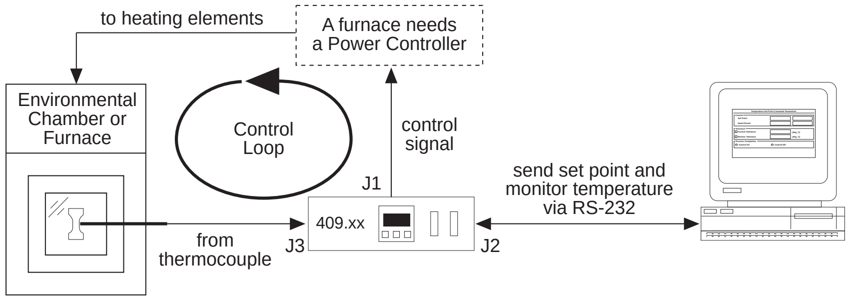
How it works

The temperature control process communicates with an MTS Series 409 Temperature Controller via RS-232 commands. Two-way communication is accomplished through one of the serial communication ports of the workstation computer and connector J2 of the temperature controller.

Continued…

How it works (continued)

The process communicates a temperature set point to the temperature controller. The temperature controller maintains the temperature control loop.

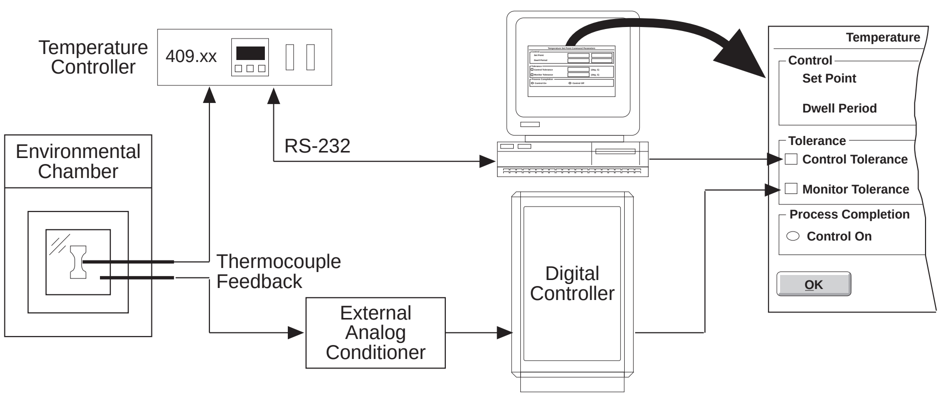


A temperature control process establishes a set point level and a dwell time. The set point must be maintained within a tolerance range for the specified dwell time before the process can end. You can create additional processes to change the set point level.

The process outputs a temperature command to the temperature controller. The temperature controller returns the current temperature (from a thermocouple) to the process for use as the control tolerance.

The monitor tolerance uses a second thermocouple with an external conditioner. The monitor feedback uses an input signal configured for an external input.

The control tolerance and the monitor tolerance use different thermocouple feedback circuits.



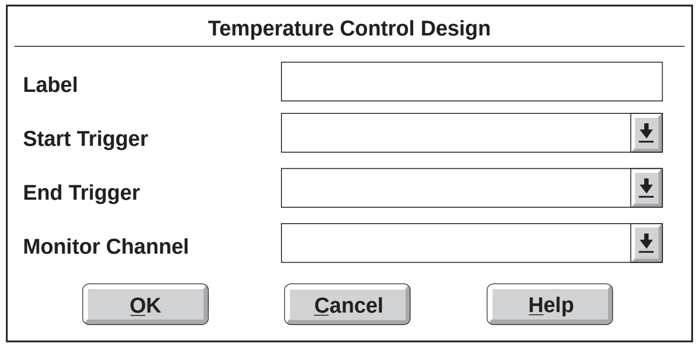
Both tolerances can be applied to the set point level at the same time. Each tolerance may be measuring different locations and may use different ranges since the temperature gradient for each thermocouple may be different.

The Processes

Temperature Control Design Window

This window names the process and specifies how the process starts and stops.

The information in this window is saved with the test template.



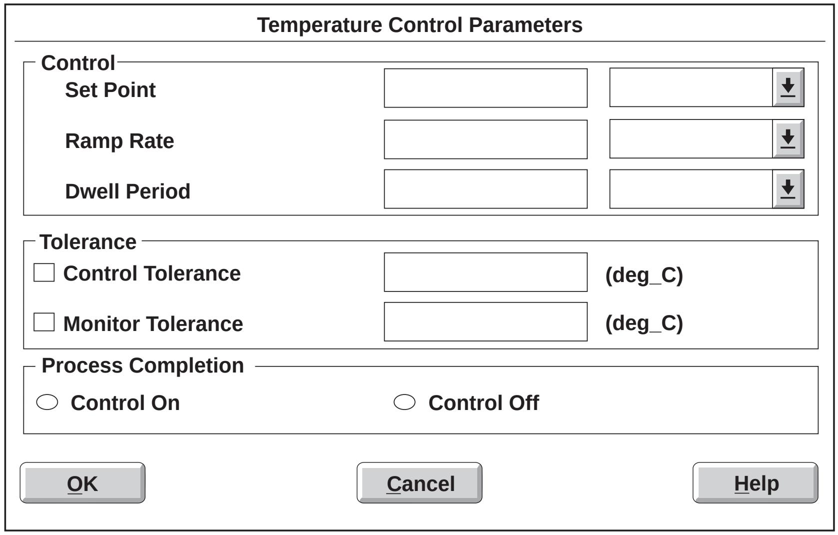
CONTROL	FUNCTION
Label	Names the process. Type the name you want to call the process in the entry field.
Start Trigger	Specifies the beginning of the process. Press the list icon and select a trigger.
End Trigger	Specifies the end of the process. Press the list icon and select a trigger.
Monitor Channel	Selects an input signal that is configured as an external channel. Pressing the list icon shows the input signal selections.

Temperature Control Parameters Window

Prerequisite

You must establish the command process with the design window before you can access the parameter window. The information in this window is saved with the test procedure.

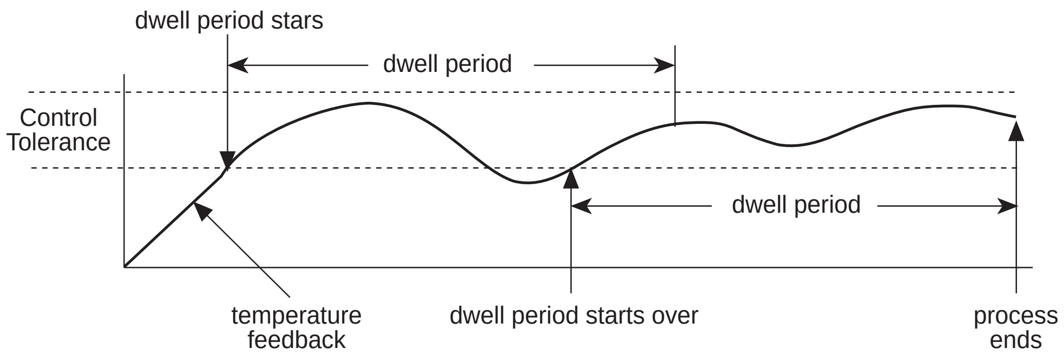
Use this window to define the parameters of a temperature control process. The two tolerance functions are enabled independently.



Control	Function
Set Point	Specifies the temperature level to be maintained by the tolerance selection. Default is 20°C.
Ramp Rate	Specifies the ramp rate to the set point.
Dwell Period	Specifies how long the temperature is maintained within a temperature tolerance range before the process ends.
Control Tolerance	Specifies a temperature tolerance range. The input to the control tolerance is from the thermocouple used by the temperature controller.
Monitor Tolerance	Specifies a monitor range. The input to the monitor tolerance is from a device independent of the temperature controller.
Process Completion	Specifies if the temperature controller maintains temperature control (control on) or not (control off) when the process is complete.

Dwell Period

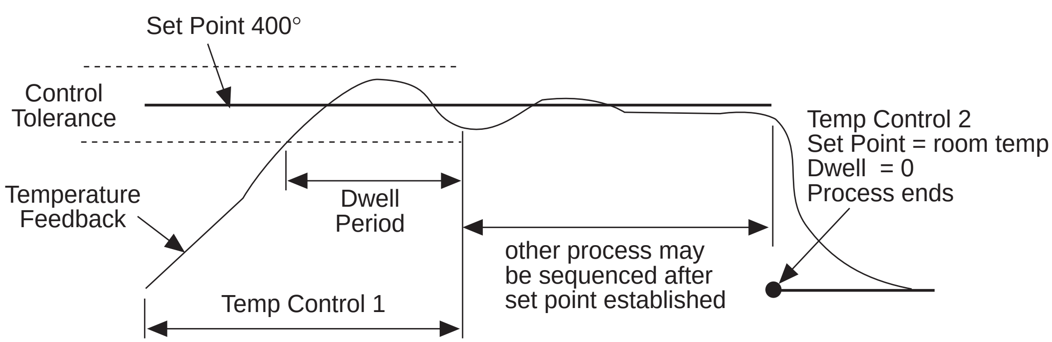
The dwell period specifies how long the set point must remain within a temperature tolerance range. One or two tolerance ranges may be used. When both the control and monitor tolerances are used, the set point must remain within both tolerance ranges or the dwell period restarts.



If the temperature exceeds the set point tolerance before the dwell period is complete, the dwell period starts over.

Process completion

Use multiple temperature control processes to manage the set point.



If the Process Completion is set to Control On, then the set point is maintained after the process is complete. To change the set point to room temperature after the process is complete, create a second process or set the Process Completion to Control Off.

Ramp rate

You can specify the ramp rate with any of the units listed. However, the temperature controller must be configured for degree per minute.

Note *You must run the COMTEST.EXE program (located in the TS2 directory) to enable the ramp rate feature. This program also displays the ramping capability of the temperature controller.*

Some control modules used in the 409 series of Temperature Controllers do not support ramping or have partial implementation.

Temperature Data Acquisition

The temperature data acquisition process is a data collection process.

Note *Pages 230 through 233 provide details common to all processes such as default settings, using triggers, and how to reach the process windows.*

Contents

Prerequisite

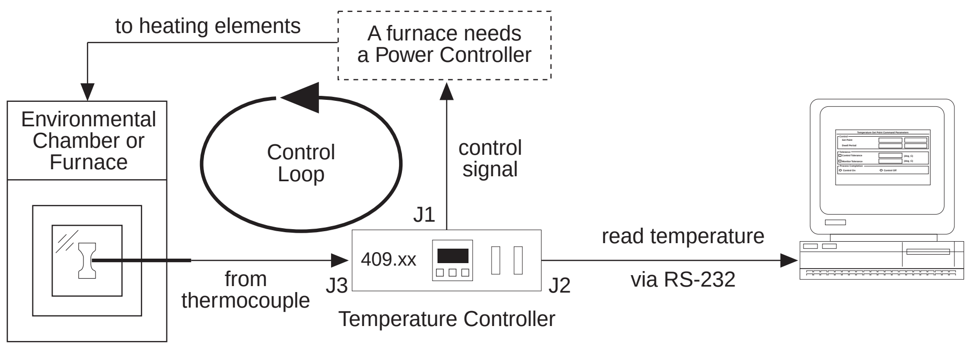
This process is designed to be used with an MTS Model 409.80 or 409.82 Temperature Controller with the RS-232 interface option (p/n 475319-01). A serial port must be specified with the hardware configuration mode of the TestStar software installation program.

How it works

It acquires temperature and data from any additional channels at set timed intervals. Data is acquired until a data buffer is full. A buffer's size and type determine how data is recorded. The temperature data acquisition process is similar to the data acquisition process. The differences are:

- It acquires temperature data from the RS-232 interface.
- It acquires data at a slower rate (every 2 seconds).

The temperature data acquisition process communicates with an MTS Series 409 Temperature Controller via RS-232 commands. Communication is accomplished through one of the serial communication ports of the workstation computer and connector J2 of the temperature controller.

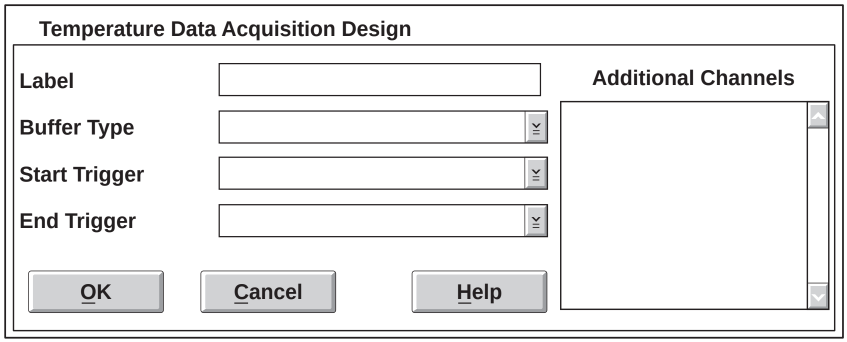


The process reads the temperature from the RS232 interface at periodically.

Temperature Data Acquisition Design Window

This window names the process and specifies how the process starts and stops.

The information in this window is saved with the test template.



CONTROL	FUNCTION
Label	Names the process. Type the name you want to call the process in the entry field.
Buffer Type	Specifies how data is temporarily stored until the data is saved to disk.
Start Trigger	Specifies the beginning of the process. Press the list icon and select a trigger.
End Trigger	Specifies the end of the process. Press the list icon and select a trigger.
Additional Channels	Selects additional input signals for data acquisition. Highlight any input signal to select it. You can acquire data from as many channels as you wish.

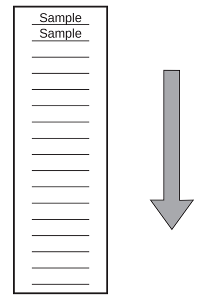
Buffer type

Because each type of buffer offers different characteristics, you may define one data acquisition process to simply trigger another data acquisition process to acquire data.

Single buffer

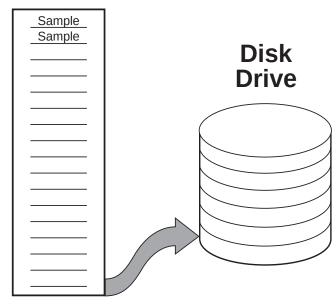
Data is recorded to fill the buffer once, then stops the process and saves the data to disk. The size of the data buffer determines how much data to collect.

This selection acquires specific data in a test, and is useful for triggering other processes.



Continuous buffer

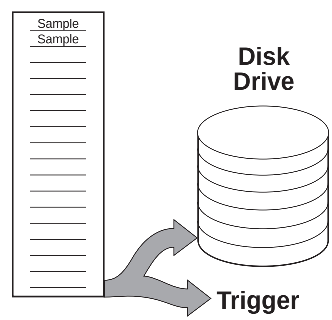
In this mode, data is continuously taken and automatically stored to disk. Storing continues until the end trigger requirement occurs or the step done process is encountered. The only limit to the total number of samples is the drive's storage capacity.



Continuous with trigger buffer

This buffer type functions the same as the *continuous* buffer except it issues a trigger each time the buffer if full. You specify the size of the buffer to determine when the trigger is issued. This process ends at the end of a step.

This selection is useful to trigger other processes at regular intervals while saving data.



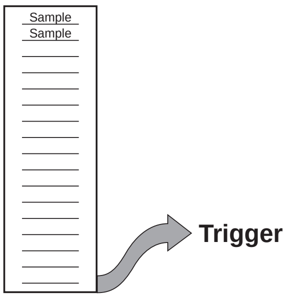
Continued…

Buffer type (continued)

Trigger only buffer

This buffer type functions the same as the *continuous with trigger* buffer except it does not save data. You specify the size of the buffer to determine when the trigger is issued. This process only ends at the end of a step.

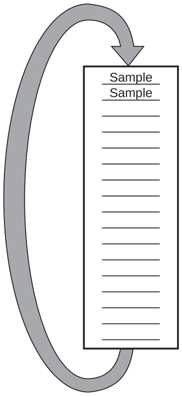
This selection is useful to trigger other processes at regular intervals without saving data.



Circular buffer

A circular buffer continuously records data to the buffer. When the buffer is full, the data is loaded into the top again, overwriting the oldest data. This continues until there is an end trigger or the step ends and saves the data to disk.

This type of buffering is useful when the data just before some event (such as specimen failure) is crucial, but data is not required for the whole test.

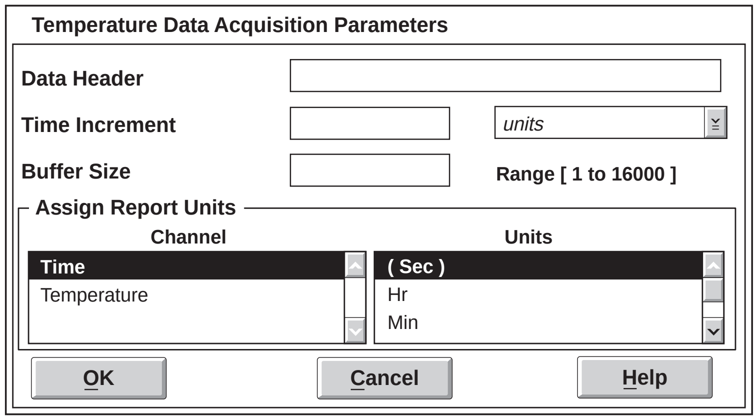


Temperature Data Acquisition Parameters Window

Prerequisite

You must establish the data acquisition process with the design window before you can access the parameter window. The information in this window is saved with the test procedure.

Use this window to define the parameters of a temperature control process. The two tolerance functions are enabled independently.



CONTROL	FUNCTION
Data Header	Labels the data in the data file.
Time Increment	Specifies how often data is acquired for all of the selected channels.
Buffer Size	Specifies the number of data elements the buffer can store. A data element represents one data sample of the temperature channel and each additional input signal channel.
Assign Report Units	This area of the window specifies the units for each master and slave channel. Select a channel, then select the units for that channel. Repeat this for each channel.

Buffer size

The default buffer size is 1024. You can set the data buffer to record 1 to 16,000 data elements. A data element includes the data from the master and slave channels along with a time stamp.

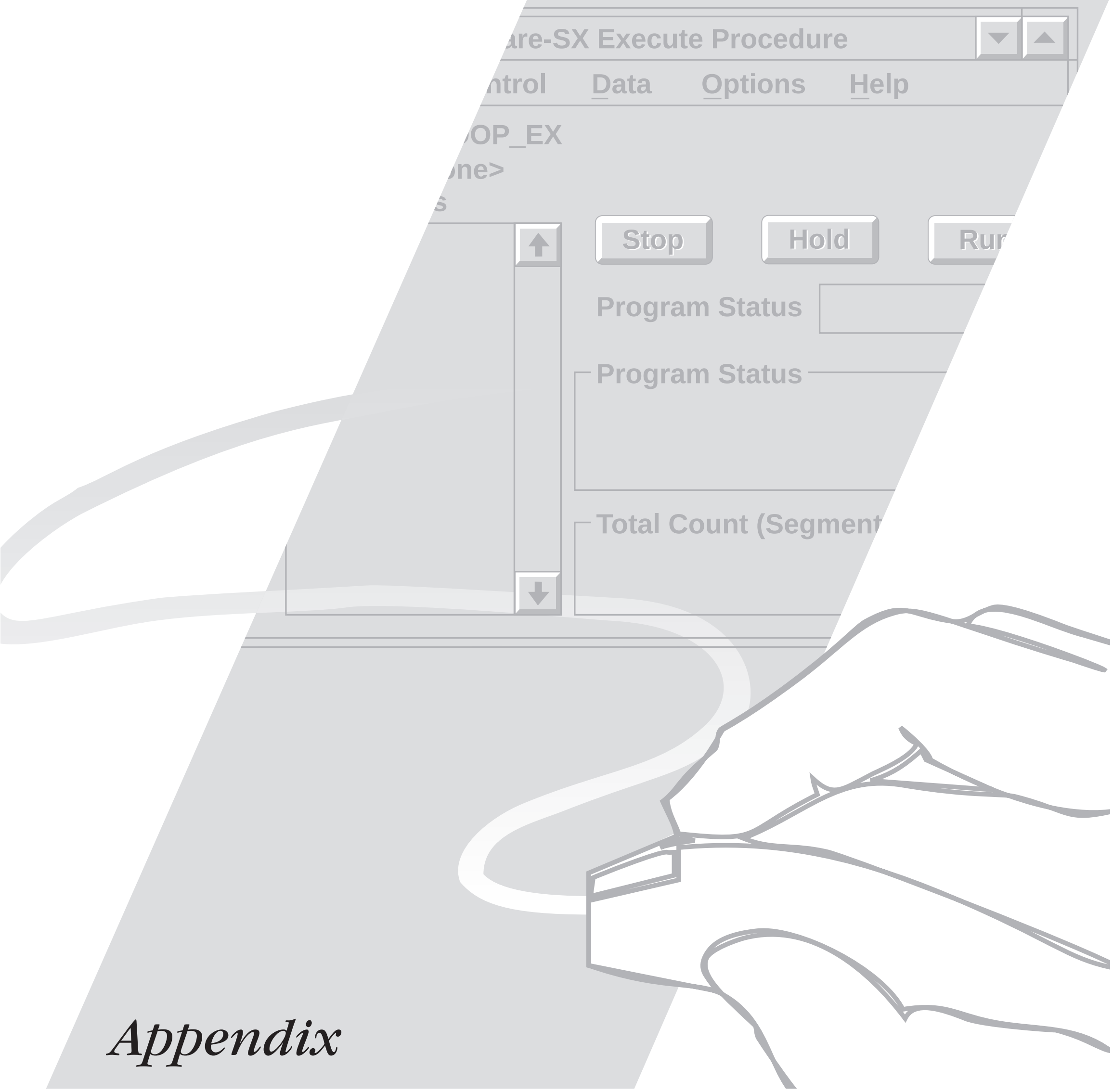
Assigning report units

Use this procedure to assign units to each master and slave channel.

1. Highlight a channel in the Channel column. This causes the Units column to display the appropriate units for the dimension of the channel.
2. Highlight the units you want assigned to the channel data.
3. Repeat this procedure for each channel.

TestStar™ II

Control System



Appendix

Appendix

Contents

Appendix

Appendix A
Examples

This chapter includes listings of the example templates and procedures that are included with the TestWare-SX software. The listings are acquired by using the Print to File feature and using a word processing program to format the file.

Note *The default procedures are test templates.*

Contents

Appendix

Fatigue Default Procedure

Procedure Name = FATIGUE Default Procedure
File Specification = C:\TS2\TWSX\fatigue.000

Data File Options
File Format = Excel Text File
Log Events = Yes
Include Procedure Description = No

Recovery Options
auto save = disabled

RAMP UP: Step
Step Done Trigger 1 = RAMP UP

RAMP UP: Monotonic Command
Start Trigger = Step Start
End Trigger = <none>
Segment Shape = Relative Ramp
Rate = 10 m/Sec
Axial
Control Mode= stroke
Endlevel = 0 mm

CYCLING: Step
Step Done Trigger 1 = CYCLING

CYCLING: Cyclic Command
Start Trigger = Step Start
End Trigger = <none>
Segment Shape = Relative Haversine
Frequency = 10 Hz
Repeats = 11110 cycles
Compensation = None
Axial
Control Mode= stroke
Endlevel 1 = 10 mm
Endlevel 2 = -10 mm

RETURN TO ZERO: Step
Step Done Trigger 1 = RAMP DOWN

RAMP DOWN: Monotonic Command
Start Trigger = Step Start
End Trigger = <none>
Segment Shape = Relative Ramp
Rate = 10 m/Sec
Axial
Control Mode= Length A SG
Endlevel = 0 m

Fatigue (Displacement)

Procedure Name = FATIGUE(DISP)
File Specification = C:\TS\TWSX\FATIGUE.001

Data File Options
File Format = Excel Text File
Log Events = Yes
Include Procedure Description = No

Recovery Options
auto save = disabled

RAMP UP: Step
Step Done Trigger 1 = RAMP UP

RAMP UP: Monotonic Command
Start Trigger = Step Start
End Trigger = <none>
Segment Shape = Ramp
Rate = 0.01 in/Sec
Axial
Control Mode = Length A SG
Endlevel = 0.04 in

CYCLING: Step
Step Done Trigger 1 = CYCLING

Continued...

Fatigue Displacement (continued)

CYCLING: Cyclic Command
    Start Trigger = Step Start
    End Trigger = <none>
    Segment Shape = Relative Haversine
    Frequency = 10 Hz
    Repeats = 1000 cycles
    Compensation = None
    Axial
        Control Mode= Length A SG
        Endlevel 1 = 0.04 in
        Endlevel 2 = -0.04 in

RETURN TO ZERO: Step
    Step Done Trigger 1 = RAMP DOWN

    RAMP DOWN: Monotonic Command
        Start Trigger = Step Start
        End Trigger = <none>
        Segment Shape = Ramp
        Rate = 0.01 in/Sec
        Axial
            Control Mode= Length A SG
            Endlevel = 0 in

Fatigue (Force)

Procedure Name = FATIGUE(FORCE)
File Specification = C:\TS\TWSX\FATIGUE.002

Data File Options
    File Format = Excel Text File
    Log Events = Yes
    Include Procedure Description = No

Recovery Options
    auto save = disabled

RAMP UP: Step
    Step Done Trigger 1 = RAMP UP

    RAMP UP: Monotonic Command
        Start Trigger = Step Start
        End Trigger = <none>
        Segment Shape = Ramp
        Rate = 50 lbf/Sec
        Axial
            Control Mode = Force A SG
            Endlevel = 600 lbf

CYCLING: Step
    Step Done Trigger 1 = CYCLING

    CYCLING: Cyclic Command
        Start Trigger = Step Start
        End Trigger = <none>
        Segment Shape = Relative Haversine
        Frequency = 10 Hz
        Repeats = 1000 cycles
        Compensation = None
        Axial
            Control Mode= Force A SG
            Endlevel 1 = 900 lbf
            Endlevel 2 = 100 lbf

RETURN TO ZERO: Step
    tep Done Trigger 1 = RAMP DOWN

    RAMP DOWN: Monotonic Command
        Start Trigger = Step Start
        End Trigger = <none>
        Segment Shape = Ramp
        Rate = 50 lbf/Sec
        Axial
            Control Mode= Force A SG
            Endlevel = 0 lbf

Fatigue w/Operator Event

Procedure Name = fatiqueOpEvt Default Procedure
File Specification = C:\TS2\TWSX\fatigueOpEvt.000
Software Version = 4.0A
Printout Date = 12:51:24 PM 10/28/96

Data File Options
    File Format = Plain Text File
    Log Events = Yes
    Include Procedure Description = No

Recovery Options
    Autosave disabled.

Set-Up: Step
    Step Done Trigger 1 = Hold

    Operator Info: Operator Information
        Start Trigger = Step Start
        End Trigger = <none>
        Form fields
            Label = Operator Name
            Default Entry =
            Type = String
            Attribute = Non-Blank

            Label = Specimen Type
            Default Entry =
            Type = String
            Attribute = Non-Blank

            Label = <end of form>
            Default Entry =
            Type = String
            Attribute = None

    Ramp Down: Monotonic Command
        Start Trigger = Operator Info
        End Trigger = Load Limit
        Segment Shape = Relative Ramp
        Rate = 1 (mm/Sec)
        Axial
            Control Mode = stoke
            End level = 0 (mm)

    Load Limit: Data Limit Detector
        Start Trigger = Operator Info
        End Trigger = <none>
        Data Channel = load
        Limit Value = -100 (N)
        Limit Value is = Absolute
        Trigger Options = Either Transition

    Hold: Hold Command
        Start Trigger = Ramp Down
        End Trigger = <none>
        Hold Time = 5 (Sec)
        Axial
            Control Mode = stoke

Cycling: Step
    Step Done Trigger 1 = Cycling
    Step Done Trigger 2 = Operator Event

    Trigger Data: Data Acquisition
        Start Trigger = Step Start
        End Trigger = <none>
        Mode = Level Crossing
        Buffer Type = Trigger only
        Master Channel = Axial Segments
        Data Header = Trigger Data
        Level Increment = 50 cycles
        Buffer Size = 1

    Loop Data: Data Acquisition
        Start Trigger = Trigger Data
        End Trigger = <none>
        Mode = Timed
        Buffer Type = Single
        Master Channel = Time
        Slave Channel 1 = Stroke
        Slave Channel 2 = load
        Slave Channel 3 = Axial Segments
        Data Header = Loop Data
        Time Increment = 0.01 (Sec)
        Buffer Size = 20

Continued…

Fatigue w/Operator Event (continued)

Peak/Valley: Data Acquisition
Start Trigger = Step Start
End Trigger = <none>
Mode = Peak / Valley
Buffer Type = Continuous
Master Channel = Stroke
Slave Channel 1 = load
Slave Channel 2 = Axial Segments
Data Header = P/V Data
Sensitivity = 1 (mm)
Buffer Size = 100

Cycling: Cyclic Command
Start Trigger = Step Start
End Trigger = <none>
Segment Shape = Relative Haversine
Frequency = 2 (Hz)
Repeats = 120 cycles
Compensation = None
Axial
Control Mode = stoke
End level 1 = 0 (mm)
End level 2 = 0 (mm)

Operator Event: Operator Event
Start Trigger = Step Start
End Trigger = <none>
Button ID = Button 1
Single Shot = Yes
Button Label = Stop Test
Description = Press now to end test.
Grab Focus = Yes

Tuning (Force-Compression)

Procedure Name = TUNING(FORCE-COMPRESSION)
File Specification = C:\TS2\TWSX\TUNING.002

Data File Options
File Format = Excel Text File
Log Events = Yes
Include Procedure Description = No

Recovery Options
auto save = disabled

TUNING-STEP WAVE: Step
Step Done Trigger 1 = CYCLING
CYCLING: Cyclic Command
Start Trigger = Step Start
End Trigger = <none>
Segment Shape = Step
Frequency = 1 Hz
Repeats = 0 cycles
Amplitude/Mean Control = Off
Axial
Control Mode= Force A SG
Endlevel 1 -50 lbf

Ramp Hold Default Procedure

Procedure Name = RAMPHOLD Default Procedure
File Specification = C:\TS2\TWSX\RAMPHOLD.000

Data File Options
File Format = Plain Text File
Log Events = Yes
Include Procedure Description = No

Recovery Options
auto save = disabled

Ramp up: Step
Step Done Trigger 1 = Ramp up

Ramp up: Monotonic Command
Start Trigger = Step Start
End Trigger = <none>
Segment Shape = Ramp
Rate = 1 (in/Sec)

Continued…

Ramp Hold Default Procedure (continued)

Axial
Control Mode= Length A SG
Endlevel = 1 (in)

Hold: Step
Step Done Trigger 1 = Hold

Hold: Hold Command
Start Trigger = Step Start
End Trigger = <none>
Hold Time = 20 (Sec)
Axial
Control Mode= Length A SG

Tuning Default Procedure

Procedure Name = TUNING Default Procedure
File Specification = C:\TS2\TWSX\TUNING.000

Data File Options
File Format = Excel Text File
Log Events = Yes
Include Procedure Description = No

Recovery Options
auto save = disabled

TUNING-STEP WAVE: Step
Step Done Trigger 1 = CYCLING
CYCLING: Cyclic Command
Start Trigger = Step Start
End Trigger = <none>
Segment Shape = Step
Frequency = 0 Hz
Repeats = 0 cycles
Amplitude/Mean Control = Off
Axial
Control Mode= Length A SG
Endlevel 1 = 0 m
Endlevel 2 = 0 m

Tuning (Displacement)

Procedure Name = TUNING(DISP)
File Specification = C:\TS2\TWSX\TUNING.001

Data File Options
File Format = Excel Text File
Log Events = Yes
Include Procedure Description = No

Recovery Options
auto save = disabled

TUNING-STEP WAVE: Step
Step Done Trigger 1 = CYCLING

CYCLING: Cyclic Command
Start Trigger = Step Start
End Trigger = <none>
Segment Shape = Step
Frequency = 1 Hz
Repeats = 0 cycles
Amplitude/Mean Control = Off
Axial
Control Mode= Length A SG
Endlevel 1 = 0.25 in
Endlevel 2 = 0 in

Appendix B
TestWare-SX Specifications

Designing A Test

Maximum number of test templates	Limited only by disk storage capacity
Maximum number of test procedures	999 per template
Maximum number of steps and processes	Limited only by computer memory
	Maximum of 10 Data acquisition processes of the same type in one step (including those used by TestStar displays).

Command

Minimum ramp time	0.001 second
Maximum ramp time	> 1000 years
Minimum frequency	0.001 Hz
Maximum frequency	1000 Hz
Maximum number of end levels (file playback)	Limited only by disk storage capacity
Maximum hold time	> 1000 years
Minimum sequencing time between processes	≈50 ms (486 cpu at 33 Mhz)
Maximum number of cycles	2.147 billion segments per command process

Data Acquisition

Minimum sampling interval	0.0002 second
Maximum sampling interval	30 minutes
Maximum number of sampled signals	no limit
Types of data acquisition	Timed, level, peak/valley, valley/peak
Types of data buffering	Single, continuous, circular, trigger only, continuous w/ trigger
Maximum data buffer size	16,000 samples per channel of data

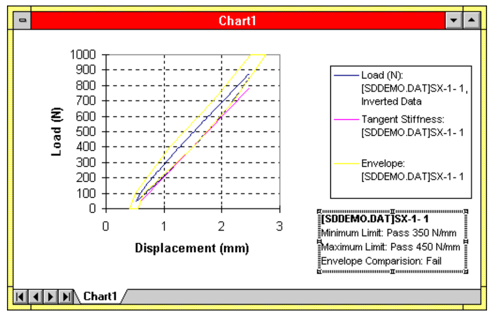
Appendix

Appendix C
Creating Plots with Excel

Enabling the TestWare-SX Excel Filter

You enable the Excel filter during the TestWare-SX software installation (see ***Installing TestWare-SX V4.0A*** on page 22). This filter ports TestWare data files (*.DAT) to Microsoft Excel, enabling Excel to quickly create plots of 790.33 test data.

Use the Excel filter to quickly create plots.



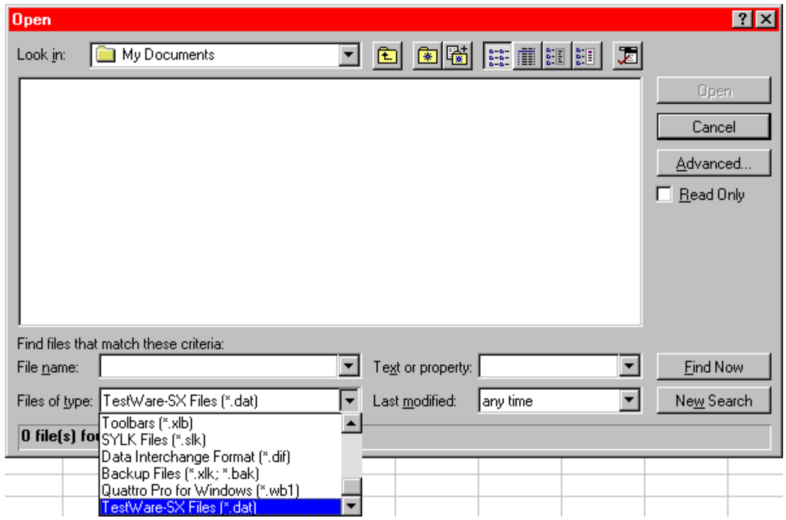
Prerequisites

- Excel 5.0 or higher for computers running Windows NT.
- A working knowledge of Excel.

If you are running Excel under a different operating system, contact MTS.

Opening Files

Use the Open File window in Excel to open a .DAT file. You must select "TestWare-SX Files (*.dat) as the file type. This opens the data file as a read-only Excel workbook. Read-only status prevents you from modifying the file (you can no longer append test data to it).



Excel opens .DAT files as read-only workbooks

CAUTION

Never open a .DAT file using Excel's file menu's Open... command.

A file opened this way can be changed and then saved under its original name. The file will stay a valid Excel file but you may not be able to add further test data to it.

Always open .DAT files through the Open TestWare-SX Data File window.

Appendix

Appendix D
Fast TestWare-SX Launch

OS/2 ONLY
This procedure can only be performed with version 3.1 of TestStar.

There may be times when you want to run a TestStar system with very little interaction. This appendix describes how to launch a specific TestWare-SX test procedure from an icon. This is accomplished with the parameters entry field in the Settings notebook. All of the parameter syntax options are also described.

The procedure creates a folder containing a TestStar configuration icon and a TestWare-SX icon connected to a specific SX template. To run a test, you open the folder, double-click on the TestStar configuration icon, wait for TestStar to open up, then double-click on the TestWare-SX icon. TestWare-SX opens to the Execution window where you can press Run to start the test.

CAUTION

The objects being created on the desktop are not fully saved until the OS/2 system is properly shutdown and cannot be backed up.

Be sure to use the Shutdown selection from the OS/2 desktop menu before turning off your computer.

Step 1 Create a new folder

This step creates a new folder in the TestStar (MTS-TSII) folder.

A Open the MTS-TSII folder.

B Select the Utility folder and press the right mouse button to display the menu for the folder.

C Select Create Another to access the sub menu and select Folder. This opens a notebook page where you can name the folder and create it.

D Type in a name for the folder in the New name entry field (for example, Fatigue Test). By default, the MTS-TSII folder should be selected. Press the Create button.

Step 2 Open the new folder

Double-click on the folder you created.

Step 3 Create a TestStar configuration icon inside the new folder

This step creates a shadow of an existing TestStar configuration file. This let's you launch the configuration from another folder.

A Open the MTS-TSII config folder.

B Highlight one of your configuration files and press the right mouse button to display the menu for the file.

C Select Create Shadow. This opens a notebook page where you can select a folder where you want to put the shadow of the configuration file.

D Highlight the folder you created in Step 1 and press the Create button.

Step 4 Create a TestWare-SX icon

A Select the TestWare-SX or TWSX icon in the MTS-TSII folder.

B Press the right mouse button to display the menu for the folder.

C Select Copy. This displays a notebook page where you can rename the select the folder you created in Step 1where you want to copy the icon.

D Rename the icon in the New name entry field (for example, Fatigue Test).

E Highlight the folder you created and press the Copy button.

Step 5 Define the TestWare-SX procedure

This step customizes how TestWare-SX is started when using the new icon. It opens a specific TestWare-SX test procedure.

A Select the TestWare-SX icon you created in Step 4 and press the right mouse button to display the menu for the icon.

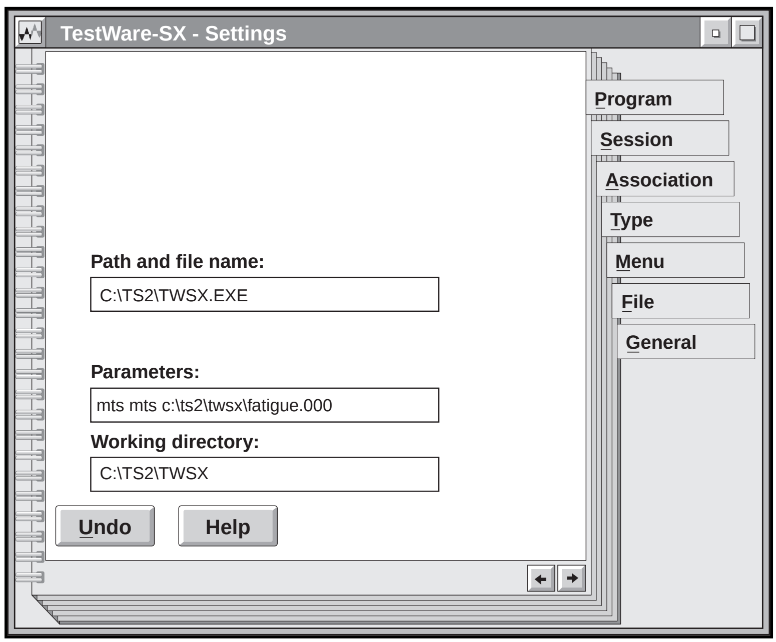
B Select Settings. This opens the Settings notebook page where you can set new parameters for the icon.

See the Syntax topic for additional options

For example, assume you want to open a template called “Fatigue” and a procedure called “Fatigue(force)”. Type the following:

c:ts2\twsx\fatigue.000 Fatigue(force) -E

This example bypasses the login (MTS MTS) and runs the Fatigue(force) procedure from the fatigue template.



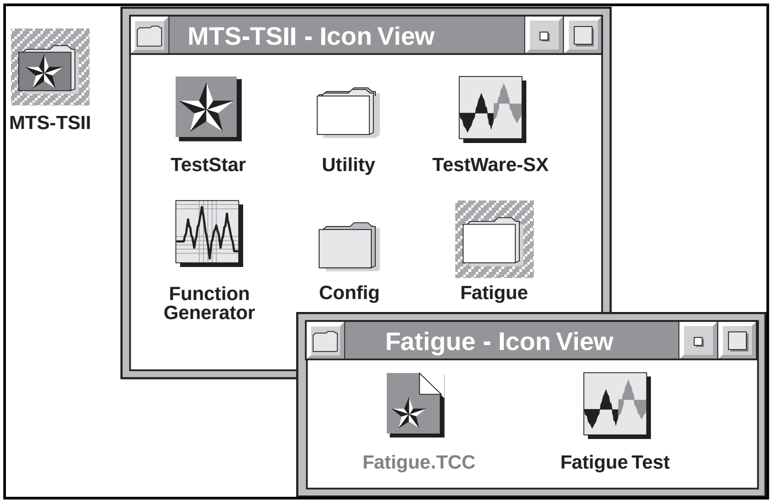
C Double-click the upper left corner icon to close the window and save the new settings.

Launching TestWare-SX

Perform the following to launch a specific test:

1. Open the MTS-TSII folder.
2. Open the Fatigue folder you created.
3. Start TestStar by double-clicking the Fatigue.TCC configuration file.
4. When TestStar is done loading, start the TestWare-SX test procedure by double-clicking the TestWare-SX application icon.

The TestStar configuration file name is gray (Fatigue.TCC). This occurs because it is a shadow.



Syntax

The following describe all of the options available for the parameters syntax.

* * template procedure -option

Where: * * represents the current username and password used to log onto TestStar. These allow you to bypass the login window when starting TestWare-SX. See Appendix E in the TestStar Reference manual for related information.

template is the complete path and filename of a test template created by TestWare-SX. This is a required parameter.

procedure is name of the template procedure. This parameter is case sensitive. If this is not specified, the template default procedure opens.

-**options** specify which TestWare-SX window to open.

The options let TestWare-SX open to different windows within the application with the specified template open. If a procedure is specified, TestWare-SX opens with the specified procedure open.

Option	Description
-T	Edit Template window
-P	Edit Procedure window
-E	Execute Procedure window
-R	Asks to open a data file

Examples

The following illustrate how the various options work.

-T

For example, **`c:ts2\twsx\fatigue.000`** **-T** starts TestWare-SX with the fatigue template open in the Edit Template window. You will be prompted to log into TestWare-SX

-P

For example, **`c:ts2\twsx\fatigue.000`** **Fatigue(DISP) -P** starts TestWare-SX with the Fatigue(DISP) test procedure open in the Edit Procedure mode. You will be prompted to log into TestWare-SX

-E

For example, **`* * c:ts2\twsx\fatigue.000`** **-E** starts TestWare-SX with the Fatigue default Procedure open in the Execute Procedure mode. You will not be prompted to log into TestWare-SX.

Index

Numerics

A

B

C

D

E

Index

F

G

H

I

L

M

N

O

P

R

S

T

Index

U

W