Getting Started with the Operator Interface

**Contents**

[Introduction 3](#_bookmark0)

[Deploying the Operator Interface in Production 3](#_bookmark1)

[Operator Interface Environment 4](#_bookmark3)

[Running the Test Program 7](#_bookmark4)

[Test Program and Data Structure 8](#_bookmark5)

[Configuration Files 11](#_bookmark6)

[Troubleshooting 13](#_bookmark7)

# Introduction

Use the operator interface (OI) to run tests, monitor test status, and save the test results to a local directory or a remote server.

# Deploying the Operator Interface in Production

Deploy the operator interface (OI) to run in a production environment, including installing the required packages, connecting the OI to the server, configuring the station/lot settings, etc.

## Installing the OI

## Procedure

**1.** Placeholder.

## Configuring a Handler/Prober Driver

## Procedure

1. In TestStand, select **Semiconductor Module>>Configure Station**.
2. In the **General** tab, select **Enable Handlder/Prober Driver (Real or Simulated)**.
3. Select an appropriate hander/prober driver from the pull-down menu.

The following table lists the supported handler/driver model(s) of each handler/prober driver.

|  |  |
| --- | --- |
| **Handler/Prober Driver** | **Supported Handler/Prober Model(s)** |
| Built-in Simulated Handler Driver | N/A. The NI Built-in Simulated Handler Driver is used for simulating handler functions when you do not have access to a real handler. |
| Accretech GPIB Prober Driver |  |
| General GPIB Handler Driver |  |
| General TTL Handler Driver |  |
| TEL GPIB Prober Driver |  |

1. Select **Configure Handler/Prober**.
2. Specify the settings for the selected handler/prober driver.

Click the **Help** button at the botton left for more information about each option.

1. Click **OK**.
2. Click **OK**.

## Enabling STDF Log for Processing Test Results

## Procedure

**Note** If this configuration is not done, the OI throws an error **Datalog does not exist. Call mechanical engineer and platform test engineer.** after you click the **Save Test Results** button.

1. In TestStand, select **Semiconductor Module>>Configure Station**.
2. In the **Advanced** tab, select **Result Processing**.
3. If the **STDF Log** option is not listed in the **Output Name** column, click the right arrow on the top right and select

**STDF Log** from the list.

1. Verify that the **STDF Log** option is listed in the **Output Name** column and place a checkmark in the **Enabled**

column.

1. Click **OK** .
2. Click **OK**.

## Connecting the OI to Server

You must connect the OI to a test program server and a test data server so that it can download test program from and upload test data to the correct locations.

## Procedure

1. Open [*Configuration Dir*]/db/env\_config.txt.
2. Update the [server\_data\_IP\_1] variable values to match the test data server you are connecting to. For example, change the *server\_data\_IP\_1* value to the IP address of your server and confirm the customer ID value is correct.
3. To connect to a second test data server for another customer, complete the following steps:
   1. Update the *server\_data\_FTP\_count* value to 2.

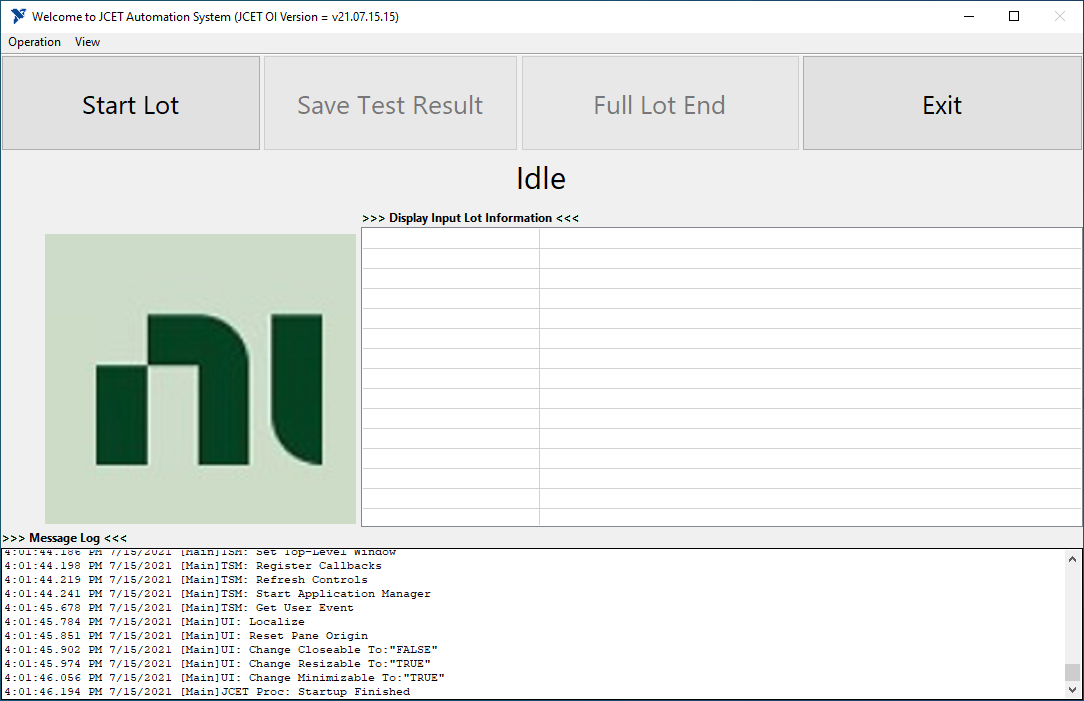
 **Note:** If the *server\_data\_FTP\_count* value is 1, the OI saves the test data to the first test data server even if you have configured a second test data server.

* 1. Update the [server\_data\_IP\_2] variable values to match the second test data server you are connecting to. For example, change the *server\_data\_IP\_2* value to the IP address of your second server and confirm the customer ID value is correct.

1. Update the [server\_pmg\_IP] variable values to match the test program server you are connecting to. For example, change the *server\_pmg\_IP* value to the IP address of your server and confirm the customer ID value is correct.

# Operator Interface Environment

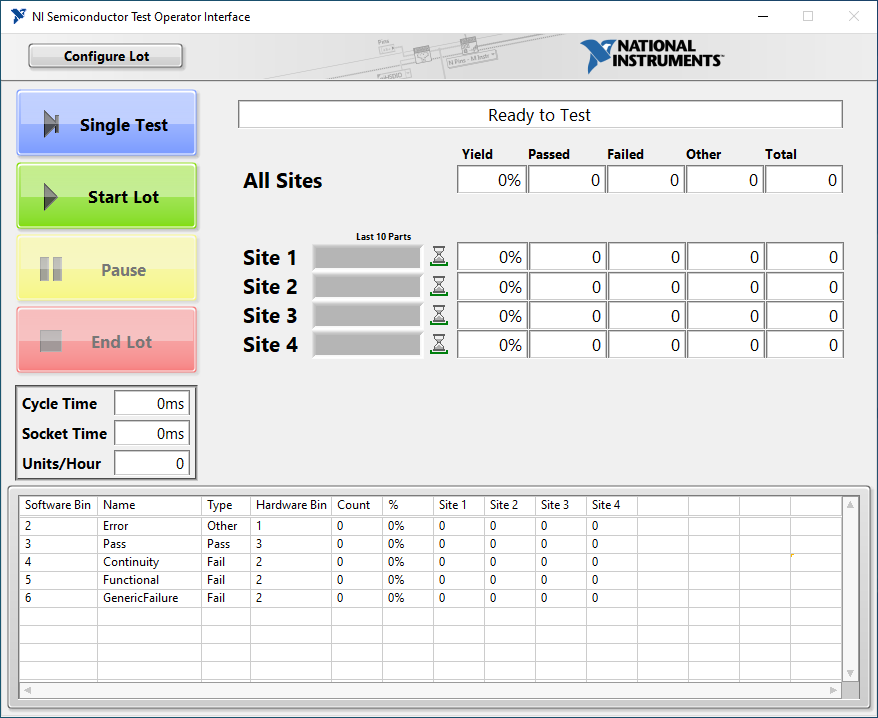
The operator interface (OI) has two user interfaces: the main panel and the default operator interface.



## Figure 1: Main Panel Table 1: Main Panel Key

|  |  |
| --- | --- |
| **Initialize Lot** | Initializes the lot by downloading the test program and entering or scanning lot information. |
| **Save Test Result** | Saves the intermediate test data to a local directory or a remote server.  The following are the default directories. To modify the locations to save the data, contact NI engineers.   * C:\testdata\DLOG\CustomerID\DevName   \Customer\_LotNo\SubLotNo\csvdir   * [*server\_data*]\CustomerID\DevName   \Customer\_LotNo\SubLotNo\csvdir |
| **Full Lot End** | Generates summary and transfer reports from the intermediate test data and saves the reports to a local directory or a remote server.  The following are the default directories. To modify the locations to save the reports, contact NI engineers.   * C:\testdata\DLOG\CustomerID   \DevName\Customer\_LotNo\SubLotNo  \FullLotFile   * [*server\_data*]\CustomerID   \DevName\Customer\_LotNo\SubLotNo  \FullLotFile |
| **Exit** | Exits the OI completely. |

|  |  |
| --- | --- |
| **Status Message** | Displays the OI status in real time. |
| **Lot Information** | Displays basic information of the current lot. The information is either manually entered by the operator or scanned from a barcode. |
| **OI Log** | Displays the OI log in real time, which is helpful for troubleshooting. |



## Figure 2: Default Operator Interface Table 2: Default Operator Interface Key

|  |  |
| --- | --- |
| **Configure Lot** | Resets the current lot to its initial state. |
| **Single Test** | Runs the test program once. |
| **Start Lot** | Runs lot testing, which runs the test program continuously until all DUTs in the lot have been tested. |
| **Pause/Resume** | Pauses or resumes test running. |
| **End Lot** | Stops running the lot. |
| **Test Status Message** | Displays the status of the tester in real time. |

|  |  |
| --- | --- |
| **Site Execution Data** | Displays statistical and status information for each site in real time. |
| **Statistics Indicator** | Displays statistical information for a lot execution, which can be used for measuring test system performance:   * **Cycle Time**—Socket time plus index time, such as the time required to perform other tasks, such as binning, placing DUTs, and generating reports. * **Socket Time**—Time it takes to perform test code or time spent by the DUT in the test socket, as measured by the time elapsed between receiving the start-of-test (SOT) notification from the handler or prober and sending the end-of-test (EOT) notification to the handler or prober. * **Units/Hour**—Number of DUTs tested per hour. |
| **Bin Table** | Displays information about the binning of DUTs for the current lot, such as the  names of the soft bins, the associated hard bins, and the DUT counts for each site. |

# Running the Test Program

## Procedure

1. Double-click the Semi OI icon on the desktop to launch the operator interface (OI).
2. When prompted, select **Operator** or **MES**, depending on whether you want to run the OI in operator mode for mass production or engineer mode for debugging. The following table lists the differences between the modes.

|  |  |  |  |
| --- | --- | --- | --- |
| **Mode** | **Verification Password Required?** | **Alarm Password Required?** | **Lot and Program Definition Source** |
| Operator | Yes | Yes | BarCodeDefinition.t |
| MES | No | Yes | .mes file |

xt

1. Click **Initialize Lot** on the main panel.
2. When prompted to enter program information, scan from a barcode to automatically pull the information or enter it manually.
   * **(Operator Mode)** Enter the **Customer ID**, then the download information **Program Name** and **Mode Code** as defined in BarCodeDefinition.txt, located at [*server\_pgm\_dir*]/NI\_STS/Product/ [*Customer\_ID*].
   * **(MES Mode)** Enter the **Sub LotNo** and **Mode Code** as defined in the .mes file, located at

[*server\_pgm\_dir*]/MesFile/[*Sub\_LotNo*]/[*Sub\_LotNo*]\_[*Mode\_Code*].mes.

1. Click **Download** to download the test program from the server.

The OI downloads the test program from the following default directory: [*server\_pgm\_dir*]/NI\_STS/ Product/[*Customer\_ID*]/[*Zipfile\_Name*], where *Zipfile\_Name* is the *Zipfile\_Name* value defined in BarCodeDefinition.txt or .mes file, depending on whether if you are running the OI in operator or MES mode.

1. After the download is complete, scan from a barcode to automatically pull lot information or enter lot information in the dialog box that appears.
   * **(Operator Mode)** Enter **Operator\_ID**, **Device\_Name**, etc, as defined in BarCodeDefinition.txt. The **Tester\_ID** field is automatically populated with the computer name, accessible by right-

clicking **My Computer** and selecting **Properties**. The other auto-populated fields are pulled from

BarCodeDefinition.txt.

* + **(MES Mode)** Enter **Operator\_ID**, **Test\_Code**, and **Test\_BinNo**.

The auto-populated fields are pulled from Mesfile.mes.

1. Click **OK**.
2. Confirm the lot information and click **OK**.
3. Click **Lot Test** on the popup window.
4. When the **Wait for SOT** dialog box appears, click **Single Test** or **Start Lot** on the default operator interface to run a single test or continuous tests.

 **Note:** If the test program runs only once after you click **Start Lot**, select **Semiconductor Module>>Configure Station** in TestStand to verify the handler settings.

1. Monitor the test status in the following ways:
   * View the real-time lot test progress in the **Test Status Message** section of the default operator interface.
   * Monitor the real-time status of lot testing by selecting **View>>Monitor Window** on the main panel.

 **Note:** NI recommends that you use this window for debugging only. For mass production, set the

*Real\_Time\_Monitor\_flag* variable in env\_config.txt to 0 to disable this window.

* + View mid-lot summary by selecting **View>>Mid-Lot Summary** on the main panel.

 **Note:** In case that the default operator interface hides or disappears, select **View>>STM-OI Window** on the main panel to bring it to the front.

1. Perform the following steps on the default operator interface when necessary.
   * To pause testing at any time, click **Pause**.
   * To resume a paused test, click **Resume**.
   * To stop testing the lot, click **End Lot**.
2. When the lot testing completes, proceed with the following steps on the main panel.
   * To view test reports, select **View>>Report** on the main panel.
   * To save the intermediate test data and upload it to the server, click **Save Test Result**.

 **Note:** Select **View>>Configuration Window** in the main panel to see where you can find the test data files locally.

* + To start the next lot, click **Initialize Lot** on the main panel and repeat the previous steps of lot testing.

 **Note:** You must save the test result before starting the next lot. If you do not click **Save Test Result**

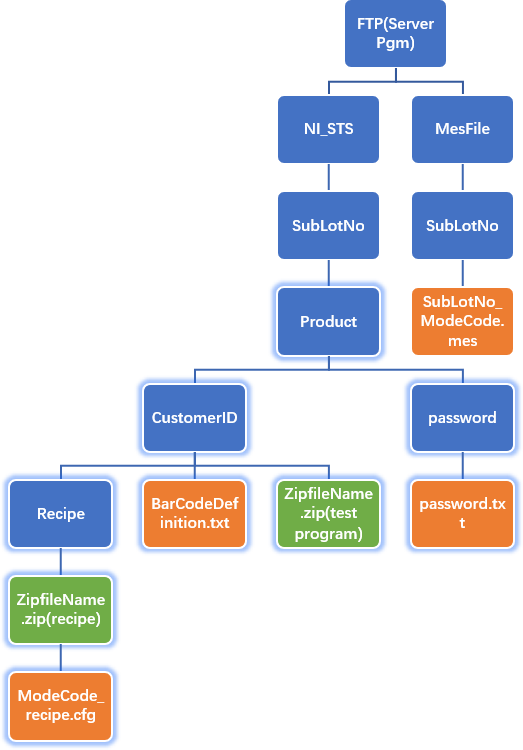
on the main panel after the lot completes, the **Initialize Lot** button on the main panel is dimmed.

* + To generate summary and transfer reports from the intermediate test data, click **Full Lot End**.
  + To exit the OI completely, click **Exit**.

# Test Program and Data Structure

By default, test program and configuration files are placed in the test program server, test data and OI log are placed in the test data server, with predefined folder structures as illustrated below, in which blue indicates folders, orange indicates files, and green indicates packaged zip files. The test data is also saved locally in C:\testdata\DLOG with the same folder structures as the test data server.

 **Note:** Some of the paths and folder names can be customized in the env\_config.txt file. To modify the location of the packaged zip files, contact NI engineers.



## Figure 3: Test Program Structure

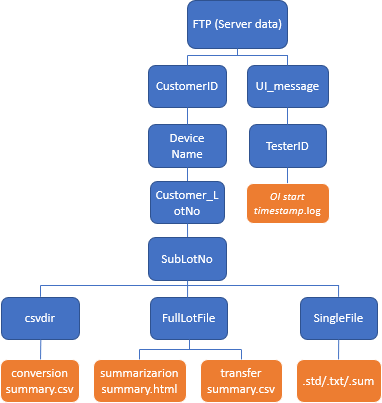
 **Note:** When packaging files or folders, place them at the top level of the package file rather than a folder with the same name as the package file. For example, use

.../*Zipfile\_Name*.zip/*Mode\_Code*\_recipe.cfg instead of .../*Zipfile\_Name*.zip/ ZipFileName/*ModeCode*\_recipe.cfg.

## Table 3: Test Program Folders and Files Details

|  |  |  |
| --- | --- | --- |
| **Folder/File** | **File Type(s) Included** | **Description** |
| [*Sub\_LotNo*]\_[*Mode\_Code*].me | sMes file | Defines the lot and program information in MES mode. |
| *Mode\_Code*\_recipe.cfg | Recipe file | Defines the following information:   * Test data paths, naming patterns, and compression type * Lot information fields that the user enters or the OI pulls from other configuration files * Test code conversion configuration * STDF attribute configuration * Bin configuration |

|  |  |  |
| --- | --- | --- |
| **Folder/File** | **File Type(s) Included** | **Description** |
| BarCodeDefinition.txt | Barcode file | Defines the following information in operator mode:   * Program information * Tester OS version * Mode code |
| *Zipfile\_Name*.zip | Zipped test program | Zipped test program for the OI to download. *Zipfile\_Name* is the *Zipfile\_Name* value defined in  BarCodeDefinition.txt or  .mes file, depending on whether you are running the OI in operator or MES mode. |
| password.txt | Password file | Defines the verification password(s) in operator mode and the alarm password. |



## Figure 4: Test Data Structure

## Table 4: Test Data Folders and Files Details

|  |  |  |
| --- | --- | --- |
| **Folder/File** | **File Type(s) Included** | **Description** |
| UIlog | .log | Records detailed operation steps, alarm information, and results. The data is saved in real time. |

|  |  |  |
| --- | --- | --- |
| **Folder/File** | **File Type(s) Included** | **Description** |
| csvdir | Conversion summary of test data (.csv) | Converted from the tester summary in .csv format, which is generated by clicking **Save Test Result**. It is used as the source for generating the summarization summary (.html). |
| [*Mode\_Code*]\_[*Tester\_ID*]\_co | nRfeicogr.dts xlott information, start and end time, version information, and conversion summary (.csv) path. It is generated by clicking **Save Test Result**. |
| FullLotFile | Summarization summary of test data (.html) | Generated by clicking **Full Lot End** which combines multiple conversion summary (.csv) files into one report according to certain calculation logic and the config information. Use this report to view test results in the production line. |
| Transfer summary (.csv) | Generated by clicking **Full Lot End** which combines multiple conversion summary (.csv) files into one report according to certain calculation logic and the config information. This report is  overwritten each time you click **Full Lot End**. Use this report to transfer in the production line. |
| SingleFile | .std, .txt, .sum | Stores the compressed tester datalog, including tester summary and STDF files generated by the tester. |

# Configuration Files

The operator interface (OI) uses the following configuration files to define the behavior. You can also customize the behavior by changing some of the configurations.

|  |  |  |  |
| --- | --- | --- | --- |
| **Configuration File** | **Description** | **Default Directory** | **Can Be Modified?** |
| env\_config.txt | Defines the following information:   * IP address of the program server and data server * Server or local test program path * Server or local test data path * Mes file path * Function call definitions * Alarm flag definitions * Other custom definitions | [*OI program dir*]/ db/env\_config.txt | No. |
| BarCodeDefinition.t | xDtefines the following information in operator mode:   * Program information * Tester OS version * Mode code   **Note:** If a test program is shared by multiple mode codes, sepatate the mode codes using underscores. For example, FT1\_QA1 means FT and QA share a program. To define multiple test programs, put each program in an independent row. | [*server\_pgm\_dir*]/ | Yes. |
|  | NI\_STS/Product/ |  |
|  | [*Customer\_ID*]/ |  |
|  | BarCodeDefinition.tx | t |
| .mes | Defines the lot and | [*server\_pgm\_dir*]/ | Yes. Can be modified in |
|  | program information in | MesFile/ | MES mode. |
|  | MES mode. | [*Sub\_LotNo*]/ |  |
|  |  | [*Sub\_LotNo*]\_[*Mode\_Co*  where *Sub\_LotNo* and | *de*].mes, |
|  |  | *Mode\_Code* represent the |  |
|  |  | corresponding values in the |  |
|  |  | .mes file. |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Configuration File** | **Description** | **Default Directory** | **Can Be Modified?** |
| [*Mode\_Code*]\_recipe. | cDfegfines the following information:   * Test data paths, naming patterns, and compression type * Lot information fields that the user enters or the OI pulls from other configuration files * Test code conversion configuration * STDF attribute configuration * Bin configuration | Each mode has a corresponding recipe file.   * **(Operator Mode)** [*server\_prog\_dir*]/ [*Customer\_ID*]/ Recipe/ [*Zipfile\_Name*]/ [*Mode\_Code*\_recipe. where *Customer\_ID*   is the customer ID you enter in the lot information  window, *Zipfile\_Name* and *Mode\_Code* represent the corresponding values in  BarCodeDefinition.   * **(MES Mode)** [*server\_pgm\_dir*]/ NI\_STS/Product/ [*Customer\_ID*]/ Recipe/ [*Zipfile\_Name*]/ [*Mode\_Code*\_recipe. where *Customer\_ID*, *Zipfile\_Name*, and *Mode\_Code* represent   the corresponding values in the .mes file. | Yes.  cfg,  txt.  cfg, |
| password.txt | Defines the verification password(s) in operator mode and the alarm password. | [*server\_pgm\_dir*]/ NI\_STS/Product/ password/ password.txt | No. |

# Troubleshooting

The following are the common errors or issues that you may run into when using the operator interface (OI), and the corresponding solutions.

## Test Data Found in Local Folder But Not on Server

First check the UI log message or the UI message file in C:\testdata\DLOG. If the upload is successful, refresh the server folder.

## Compression Unsuccessful with Selected Compression Type

First check the notes of the configuration file to determine whether the test program supports this compression type. If it is supported, copy the 7-Zip folder in the 7-Zip installation directory to C drive.

## Chinese Characters in OI

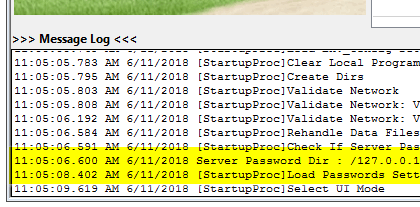
If the OI shows Chinese labels for controls and indicators, check the language settings in TestStand by selecting

## Configure>>Station Options>>Localization.

## Common Alarm Codes

|  |  |  |  |
| --- | --- | --- | --- |
| **Error/Alarm Code** | **Description** | **Possible Reason** | **Solution** |
| U01 | Cannot connect to server. Call mechanical engineer. | The server is not configured correctly. | Verify that the configuration for the test program and test data servers is correct in [*OI exe dir*]/db/ env\_config.txt.  You can verify whether  the configuration is correct by entering the IP address with the following format in the address bar in Windows Explorer: ftp://  username:password@ip For example, ftp:// [user:password@127.0.](mailto:password@127.0) |
| U02 | Datalog does not exist. Call mechanical engineer and platform test engineer. | STDF log is not enabled for processing test results. | Follow the instructions in [Enabling STDF Log for](#_bookmark2) [Processing Test Results](#_bookmark2) to enable STDF log. |
| U03 | Cannot connect to server. Call mechanical engineer. |  |  |
| U05 | Summary does not exist. Call manufacturing leader. |  |  |
| U06 | Cannot connect to server. Call mechanical engineer. |  |  |
| 7zipNotInstalled | 7zip is not installed. Install 7zip application and config path option in environment. |  |  |
| O01 | Password failed. |  |  |
| U04 | Config does not exist. Call manufacturing leader. |  |  |

:port. 0.1:21.



|  |  |  |  |
| --- | --- | --- | --- |
| **Error/Alarm Code** | **Description** | **Possible Reason** | **Solution** |
| M01 | Mesfile does not exist in server. Call IT. | * Program or configuration file is placed in the wrong location in the program server * FTP permission issues. | * Verify that the program or configuration file is placed in the correct location in the program server. The OI log displays the server location where the program attempts to read the test program or configuration file. The following image shows an example path of password.txt that the OI attempts to read from. * If the path is found on the server but the test program or configuration file   does exist, check FTP permissions. |
| M02 | Program does not exist in server. Call IT. |
| O02 | Barcodefile does not exist in server. Call IT. |
| O05 | Program does not exist in server. Call platform test engineer. |
| M03 | Mesfile modecode check failed. Call IT. |  |  |
| M04 | Mesfile program definition error. Call platform test engineer. |  |  |
| M05 | Current program does not match last program. Call platform test engineer. |  |  |
| M06 | Recipe file does not exist. Call platform test engineer. |  |  |
| M9 | Tester\_ID check failed. Call manufacturing leader. |  |  |
| M10 | Lot information check failed. Reinput values. |  |  |
| M11 | Testcode input failed. Reinput values. |  |  |
| M12 | Device\_name does not match. Call mechanical engineer setup. |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Error/Alarm Code** | **Description** | **Possible Reason** | **Solution** |
| M13 | Tester software version check failed. Call plaform test engineer. |  |  |
| M14 | SubLotNo check failed. Called IT. |  |  |
| M15 | Test program file does not exist. Call platform test engineer. |  |  |
| O03 | Program name does not exist in barcodefile. Call platform test engineer. |  |  |
| O04 | More than one program name exists in barcodefile. Call platform test engineer. |  |  |
| O06 | Barcodefile modecode check failed. Call manufacturing leader. |  |  |
| O07 | Barcodefile program- defined error. Call platform test engineer. |  |  |
| O08 | Current program does not match last program. |  |  |
| O09 | Recipe file does not exist. Call platform test engineer. |  |  |
| O10 | Tester ID check failed. Call manufacturing leader. |  |  |
| O11 | Lot information check failed. Reinput values. |  |  |
| O12 | Testcode input failed. Reinput values. |  |  |
| O13 | Device name does not match. Call mechanical engineer setup. |  |  |
| O14 | Tester software version check failed. Call platform test engineer. |  |  |
| O15 | Input2&input3 program name failed. Reinput values. |  |  |
| O16 | Test program file does not exist. Call platform test engineer. |  |  |
| 550 |  | Permission issues of file or folder in FTP server or local machine. | Check permission of file or folder in FTP server or local machine. |

ip:port.

|  |  |  |  |
| --- | --- | --- | --- |
| **Error/Alarm Code** | **Description** | **Possible Reason** | **Solution** |
| -15425 | Failed to establish FTP connection. | * The server is not configured correctly. * Firewall or anti-virus software may cause FTP connection failure. | * Verify that the configuration for the test program and test data servers is correct in [*OI exe dir*]/db/ env\_config.txt. You can verify whether the configuration is correct by entering the IP address with the following format in the address bar in Windows Explorer: ftp://   username:password@ For example, ftp:// user:password@127.   * Try turning off firewall or anti-virus software. |
| -17500 | Operation Failed.  Location: Step ‘UUT Report’ of Sequence ‘Module Plugin – UUT Done’ in ‘xxxx.seq’. Test Socket 0. | This error is from NI’s Model Plug-in. The features customized by some manufacturers, such as customized reports, are implemented in the Model Plug-in. | Contact NI engineers. |
| -1074118575 | Location: Step ‘XXX’ of sequence ‘MainSequence(or ProcessSetup or ProcessCleanup)’ in  ‘YYY.seq’. Test Socket 0. User-defined error code. Site: 0.  niDCPower/niDigital/ RFmx/niRFSG... | Test program error. | Contact NI engineers. |
| -1073807304 | Location: Step ‘Wait for SOT’ of sequence **StartOfTest** in "Seiko  Epson\_160831.seq"  User-defined error code. Site: N/A. | This error comes from Seiko  Epson\_XXX.seq, an NI handler driver program. | Contact NI engineers. |

0.0.1:21.

If the error is not listed above or if the listed solution does not work, refer to OperatorInterfaceErrors.log in the OI program directory for detailed error messages that occurred when the OI ran, including errors that occurred in OI, handler driver, and test program. To seek for NI assistance, include the OperatorInterfaceErrors.log file and OI log to help NI engineers locate errors.