



The first name in materials testing

Horizon Remote Interface

For System and Process Automation Developers



The Tinius Olsen Horizon Remote Interface or API gives system and process automation developers the tools and ability to efficiently integrate a Tinius Olsen materials testing platform into their automated process. This facilitates bi-directional interactions covering both data flow and physical actions and responses with and from the Tinius Olsen Horizon materials testing software and hardware.



Horizon Remote Interface

Namespace Index

PACKAGES

Here are the packages with brief descriptions (if available):

HorizonRemoteInterface

Class Index

CLASS LIST

Here are the classes, structs, unions and interfaces with brief descriptions:

HorizonRemoteInterface.HorizonRemoteInterface (Remote API provided as a managed .NET Framework DLL)

Namespace Documentation

HORIZONREMOTEINTERFACE NAMESPACE REFERENCE

Classes

- **class HorizonRemoteInterface**
Remote API provided as a managed .NET Framework DLL.

Class Documentation

HORIZONREMOTEINTERFACE.HORIZONREMOTEINTERFACE CLASS REFERENCE

Remote API provided as a managed .NET Framework DLL.

PUBLIC TYPES

- **enum Status { Idle, Ready, Testing, Returning, Stopped, Disabled }**
A status enumeration to support host specimen handling logic and process.

PUBLIC MEMBER FUNCTIONS

- **Boolean SendResultsAndValuesToHorizon (List< Tuple< string, string > > identifiersAndDimensions)**
Function used to send results, by Name, to Horizon. Results will have been setup previously within Horizon. If all results found and values accepted the function will return true, otherwise false is returned.
- **List< Tuple< string, string > > GetResultsAndValuesFromHorizon (List< string > results)**
Function used to retrieve results, by Name, from Horizon. A list of result names is provided to Horizon and a <string,string> Tuple is returned with the calculated/stored values.

- **Boolean SendLimitValuesToHorizon (List< Tuple< string, string, string > > limitHiLoValues)**
Function used to send high and low limit values to Horizon. These limit values may be used by Horizon to automatically flag pass or fail tests and respond appropriately. A list of result names with their high and low limit values is provided to Horizon.
- **Boolean SwitchMethod (String methodName)**
Function used to switch the method in Horizon. Typically, before or just after a specimen is successfully loaded into the machine this method is called to switch the method currently loaded into Horizon.
- **Boolean StartTest ()**
Starts the test.
- **void StopTest ()**
Stops the test. Used to interrupt the test, otherwise the test will stop once the final segment is reached or the specimen breaks.
- **void BeginReturningMachine ()**
Begins the machine return process. An automated system could carry on performing functions while the machine returns or watch the status until the machine has completely returned before proceeding.
- **void SetUpperGripState (Boolean isOpen)**
Set the upper grip state to Open or Close. Only machines with the appropriate hardware support this function.
- **void SetLowerGripState (Boolean isOpen)**
Set the lower grip state to Open or Close. Only machines with the appropriate hardware support this function.
- **Boolean IsMachineReady ()**
Verifies that the machine is ready for a specimen to be loaded and provides a point prior to specimen loading for an operator to potentially review the machine and resolve issues that may be present.
- **Status GetState ()**
Gets the current state of the testing machine.
- **String GetLastError ()**
Returns the last error that is relevant to the remote interface functionality.

DETAILED DESCRIPTION

Remote API provided as a managed .NET Framework DLL. This class provides an interface to an already running instance of Horizon. The class will support 4 main functions within Horizon. These are:

1. Import and Export functionality. Result values may be sent to Horizon and will be applied to the next specimen about to be tested. Post test, calculated results may be retrieved.



2. Method Select. Previously configured, existing test methods within Horizon may be selected and loaded in for the test that is about to be performed.
3. Test Interactions. Start Test, Stop Test, and Return Machine functions are provided to control the various steps of the test or respond to mid-test events. Also, built-in machine IO functions are supported. For example, a machine with hydraulic grips will support the Grip Open & Close functions.
4. State Information. A watchdog function, and status & error information is provided.

Future functionality:

1. Tests will be saved in Horizon's database by default. An option could be provided to never save data locally and Horizon will only perform tests in memory and transmit results back to the host system.
2. Horizon testing screen to auto clear after a specified number of tests.

MEMBER FUNCTION DOCUMENTATION

- **String HorizonRemoteInterface.**
HorizonRemoteInterface.GetLastError ()

Returns the last error that is relevant to the remote interface functionality.

Returns

The most recent error that has occurred.

- **List< Tuple< string, string > > HorizonRemoteInterface.**
HorizonRemoteInterface.
GetResultsAndValuesFromHorizon (List< string > results)

Function used to retrieve results, by Name, from Horizon. A list of result names is provided to Horizon and a <string,string> Tuple is returned with the calculated/stored values.

Parameters

results A string list of result names

Returns

A <string,string> Tuple is returned with the name and calculated/stored value.

- **Status HorizonRemoteInterface.**
HorizonRemoteInterface.GetState ()

Gets the current state of the testing machine.

Returns

The current status.

- **Boolean HorizonRemoteInterface.**
HorizonRemoteInterface.IsMachineReady ()

Verifies that the machine is ready for a specimen to be loaded and provides a point prior to specimen loading for an operator to potentially review the machine and resolve issues that may be present.

Returns

- **Boolean HorizonRemoteInterface.**
HorizonRemoteInterface.
SendLimitValuesToHorizon (List< Tuple< string, string, string > > limitHiLoValues)

Function used to send high and low limit values to Horizon. These limit values may be used by Horizon to automatically flag pass or fail tests and respond appropriately. A list of result names with their high and low limit values is provided to Horizon.

Parameters

limitHiLoValues A list of result names, high, and low values.

Returns

True, if all result names resolved and their respective high and low limits are updated successfully, otherwise false is returned.

- **Boolean HorizonRemoteInterface.**
HorizonRemoteInterface.
SendResultsAndValuesToHorizon (List< Tuple< string, string > > identifiersAndDimensions)

Function used to send results, by Name, to Horizon. Results will have been setup previously within Horizon. If all results found and values accepted the function will return true, otherwise false is returned.

Parameters

identifiersAndDimensions A tuple list of names and their associated values.

Returns

True, if all result names resolved and value updates are successful, otherwise false is returned.

- **void HorizonRemoteInterface.**
HorizonRemoteInterface.SetLowerGripState (Boolean isOpen)

Set the lower grip state to Open or Close. Only machines with the appropriate hardware support this function.

Parameters

isOpen True to open the grips, false to close the grips.

- **void HorizonRemoteInterface.**
HorizonRemoteInterface.SetUpperGripState (Boolean isOpen)

Set the upper grip state to Open or Close. Only machines with the appropriate hardware support this function.

Parameters

isOpen True to open the grips, false to close the grips.

- **Boolean HorizonRemoteInterface.**
HorizonRemoteInterface.StartTest ()

Starts the test.

Returns

True, if the method exists and was loaded without error.

- **Boolean HorizonRemoteInterface.**
HorizonRemoteInterface.SwitchMethod (String methodName)

Function used to switch the method in Horizon. Typically, before or just after a specimen is successfully loaded into the machine this method is called to switch the method currently loaded into Horizon.

Parameters

methodName The name of the method to be loaded.

Returns

True, if the specified method is found and loaded properly, otherwise false.

Tinius Olsen



www.tiniusolsen.com

- Horsham, PA, USA
- Redhill, Surrey, UK
- Noida, UP, India
- Shanghai, PR China