

# PathWave Test Sync Executive



# Contents

<b>PathWave Test Sync Executive (KS2201A) Release Notes</b> .....	<b>3</b>
<b>Release 2023B version information</b> .....	<b>4</b>
New Features and Enhancements .....	4
Breaking Changes: .....	5
Fixes .....	5
Known Issues or Limitations .....	5
<b>Release 2023 version information</b> .....	<b>7</b>
New Features .....	7
Breaking Changes: .....	7
Fixes .....	8
Known Issues or Limitations .....	8
<b>Release 2022 version information</b> .....	<b>9</b>
New features .....	9
Fixes .....	9
<b>Release 2021 version information</b> .....	<b>10</b>
New features .....	10
Fixes .....	10
<b>Release 2020 Update 1.1 version information</b> .....	<b>11</b>
New features .....	11
Fixes .....	11
<b>Release 2020 Update 1 version information</b> .....	<b>12</b>
New features .....	12
Breaking changes .....	13
Fixes .....	13
<b>Release 2020 Update 0.2 version information</b> .....	<b>14</b>
Fixes .....	14
<b>Release 2020 Update 0.1 version information</b> .....	<b>15</b>
Fixes .....	15
<b>Release 2020 version information</b> .....	<b>16</b>
Features .....	16
Known Issues or Limitations .....	17

# PathWave Test Sync Executive (KS2201A) Release Notes

For licensing information see:

- Chapter 3 (Installing Licenses) in the [PathWave Test Sync Executive User Manual](#).
- [Licensing Quick Start Guide](#).

## Release 2023B version information

PathWave Test Sync Executive 2023B continues to improve the API and adds capability to optimize and simplify creation of applications that require tight synchronization and real-time control and feedback. New features include extended Fast Data Sharing capability to share data across FPGA Sandboxes and HVI-Engine registers of different instruments, new Sync-Delay statement that together with the existing Sync-Registers and Sync-For statement enables a speed-up of multi-instrument real-time looping with Variable execution timing, new capability to query from software in real-time the resynchronization latency, and general operation enhancements and fixes.

Build #	3.27.4
Released Date	Oct 18th, 2023
Operating Systems	Microsoft Windows 10 64-bit Pro and Enterprise
Supported PXIe Infrastructure <sup>(1)</sup>	M901x, M904x, M9032/3A
Supported Instruments <sup>(1)</sup>	M310x, M320x, M330x, M530x, M520x
File Name	PathWave-TestSyncExecutive-2023B-winx64.exe

<sup>(1)</sup> For information on the specific instrument models supported and the software and firmware version requirements, see: [Instrument Software and Firmware Requirements for KS2201A](#).

## New Features and Enhancements

- New ***SyncDataSharing*** statement that enables real-time data transfer across FPGA Sandboxes, FDS-ports, and HVI-Engine registers (local and Sync registers). SyncFpgaDataSharing has been DEPRECATED and Keysight recommends you to replace it with this new statement.
- New ***SyncDelay*** statement that enables execution for synchronous delay in all instruments using a literal or Sync-Register. Together with Sync-Register and Sync-For statements, it enables implementation of better performance looping with Variable execution delays without incurring into the execution time overhead of the triggered-synchronization.
- New ***TseServiceClient*** class that enables you to create a connection to a TSE Service and query its state.

- An extension to the SyncMultiSequenceBlock statement to enable storage of triggered-synchronization latency, and a new API in the Hvi instance to query from software, this resynchronization latency in real-time during sequence execution.
- Improved system initialization and alignment operation, including a new initialization mode that ignores missing/wrong alignment calibration and enables you to run the system anyway.
- Added Fast Data Sharing real-time error detection and reporting when executing *SyncDataSharing*.
- Enhancements in TSE Service operation including simulation and support for KDI 3.2 and newer.
- Enabled to run an Hvi instance without Sync Resources when Sequencing is not used and no HVI Engines are added to the system definition.

## Breaking Changes:

- Changed return type when returning time values to always use the *Time::Duration* type instead of other types like **double**, **int** or **long long**. *Time::Duration* exposed a new property to retrieve a **double** with the value in nanoseconds.

## Fixes

- Potential high CPU usage.
- Support for strong name in .Net Assembly.
- Improve error handling and reporting.

## Known Issues or Limitations

- In systems with a large number of M5xxxA instruments, some M5xxxAs may take too long to initialize. The PXI Resource Manager may not discover all of the instruments in time and include them in the pxiesys.ini file. This results in TSE Service skipping some instruments if autodetect is used without any error or warning. For System Synchronization Modules, TSE Service will generate an error and retry.
  - If this issue is encountered, you must trigger the PXI Resource Manager to rescan the system after all the M5xxxAs have completed initialization, and restart TSE Service to re-autodetect instruments. See M5xxxA documentation for more details on boot-up timing.
  - Note that if TSE Service is configured without autodetect and instead lists instruments explicitly, TSE Service will generate an error and retries until all PXI instruments are properly listed in pxiesys.ini
- Resources stay locked if a user application crashes after the **loadToHw()** and before the **releaseHw()** while using TSE Service to manage local/remote resources. TSE Service exposes a command-line

option to force the release of resources and recover from this situation, without having to restart the system. See the *PathWave Test Sync Executive System Setup Guide* for more details.

- A single SyncDataSharing statement may not be able to support multiple transactions where an instrument is used in one transaction as a source, and as a destination in others. If that happens, the user will need to break their transactions into multiple SyncDataSharing statements.
- FDS errors generated by an incorrect operation of the User FPGA-Sandbox IP are sporadically not reported.
- Intermittent crash when running back-to-back SyncDataSharing applications, closing and opening the complete application on each execution, in Multi-Host Leader-Follower configurations.

## Release 2023 version information

PathWave Test Sync Executive 2023 continues to improve the API and adds capability to optimize and simplify creation of applications that require tight synchronization and real-time control and feedback. New features include TSE Service which simplifies and speeds up system configuration and initialization for single and multi-host applications, new Sync-Registers and Sync-For Statements to speed-up multi-instrument real-time looping, clock skew monitoring for more robust multi-chassis system initialization and operation, extended Fast Data Sharing support for configurable read-latency in user IP. In terms of licensing, we have enhanced the ordering process to simplify extending an existing system with additional TSE Licenses.

Build #	3.19.4
Released Date	May 31st, 2023
Operating Systems	Microsoft Windows 10 64-bit Pro and Enterprise
Supported PXIe Infrastructure <sup>(1)</sup>	M901x, M904x, M9032/3A
Supported Instruments <sup>(1)</sup>	M310x, M320x, M330x, M530x, M520x
File Name	PathWave-TestSyncExecutive-2023-winx64.exe

<sup>(1)</sup> For information on the specific instrument models supported and the software and firmware version requirements, see: [Instrument Software and Firmware Requirements for KS2201A](#).

## New Features

- TSE Service to automate system initialization and enable transparent multi-host operation.
- Clock Skew Monitoring based on M9032A/M9033A for robust instrument alignment on large-scale multi-chassis systems.
- Sync-Register and Sync-For statements for real-time sequencing. Speeds up synchronized multi-instrument real-time looping.
- Support for configurable read-latency in FDS ports in User IP in FPGA Sandboxes.
- User-defined PXIe-DSTAR trigger routing.

## Breaking Changes:

- Renamed TSE libraries and namespaces to include "tse" instead of "hvi" in the name:
  - Python: keysight\_hvi was renamed to keysight\_tse
  - .NET: Keysight.Hvi was renamed to Keysight.Tse

## Fixes

- Added instrument locking when running quick/optional alignment. This could have resulted in conflict and unexpected behavior when two different processes were running initialize, creating a sequencer, or calling loadToHw(), involving the same instruments concurrently.

## Known Issues or Limitations

- In systems with a large number of M5xxxA instruments, some M5xxxAs may take too long to initialize, so that PXI Resource Manager may not discover all instruments in time and include them in the pxiesys.ini file. This results in TSE Service skipping some instruments if autodetect is used without any error or warning, for System Synchronization Modules, TSE Service will generate an error and retry.
  - If this issue is encountered, you must trigger the PXI Resource Manager to rescan the system once all M5xxxAs have completed initialization, and restart TSE Service to re-autodetect instruments. See M5xxxA documentation for more details on boot-up timing.
  - Note that if TSE Service is configured without autodetect and instead lists instruments explicitly, TSE Service will generate an error and retries until all PXI instruments are properly listed in pxiesys.ini
- Detection of real-time errors during Fast Data Sharing (FDS) transaction is not available in this release. This affects real-time data transactions performed using the SyncFpgaDataSharing statement. Because of this limitation, users will not be notified by TSE if there is an error in the User FPGA IP response during an FDS transaction.
- Resources stay locked if a user application crashes after the `loadToHw()` and before the `releaseHw()` while using TSE Service to manage local/remote resources. TSE Service exposes a command-line option to force the release of resources and recover from this situation, without having to restart the system. See the *PathWave Test Sync Executive System Setup Guide* for more details.
- System synchronization procedure may fail in a multi-chassis system when **all** the following conditions apply:
  - Chronos or Chassis clock source is used as clock reference of the system
  - The primary chassis (primary is the chassis where the SyncModule has no upstream SystemSync connection) number is different than 1.
  - The first instrument that is opened by the user application does not belong to the primary chassis
  - Workaround: Try to open first an instrument that belongs to the primary chassis in the user application.
- TSE Service does not support KDI 3.2 or newer.



## Release 2022 version information

Build #	2.7.7
Released Date	July 1st, 2022
Operating Systems	Microsoft Windows 10 64-bit Pro and Enterprise
Supported PXIe Infrastructure (2)	M901x, M904x (1), M9032/3A
Supported Instruments (2)	M310x, M320x, M330x, M530x (1), M520x (1)
File Name	PathWave-TestSyncExecutive-2022-winx64.exe

(1) New products supported from 2022 release.

(2) For information on the specific instrument models supported and the software and firmware version requirements, see: [Instrument Software and Firmware Requirements for KS2201A](#).

### New features

- Multi-process support, compatible with *Keysight Distributed Infrastructure (KDI)* and Instrument's Remote Drivers.
- Support for the new M904x PXIe chassis family including the optional *High Performance Reference Clock Source (HPRCS)* and the configuration of the clock outputs.
- Extended system definition and initialization to include analog reference clocking configuration and system calibration.
- *Fast data sharing (FDS)* across instruments' FPGA sandboxes using PXIe DSTARB/C and M903x SSM together with System Sync for multi-chassis.
- Support for user-defined trigger routing across instruments and chassis.
- Logging capability to simplify troubleshooting.

### Fixes

- Fixed an issue loading the correct DLL for Keysight M9005A PXIe Chassis that is required to use simulated chassis instead.
- Improved system initialization and configuration timing.

## Release 2021 version information

Build #	1.15.7
Released Date	December 13th, 2021
Operating systems	Microsoft Windows 10 64-bit Pro and Enterprise
Supported Hardware <sup>(1)</sup>	M9032A, M9033A, M3100A, M3102A, M3201A, M3202A, M3300A, M3302A, M5302A
File Name	PathWaveTSE_2021_winx64.exe

<sup>(1)</sup> For information on the specific product models supported and SW&FW version requirements refer to: [Instrument Software and Firmware Requirements for KS2201A](#)

### New features

- New PXIe multi-chassis support based on the new M9032A/33A PXIe System Sync Modules, which deliver better scalability with faster Sync timing.
  - Support for M9031A has been discontinued. This release does not work with M9031A.
- Change in the SW licensing policy: -HVx instrument hardware options are no longer required and instead a KS2201A SW license is required per instrument or module used with Test Sync Executive.
  - For Instruments or modules that still use the -HVx hardware option, licensing will work as in previous releases. Both types of instruments or modules can work together in the same system. Test Sync Executive software will accommodate both situations and use the KS2201A software licenses for modules without the -HVx option and use the -HVx hardware option when that is available.

### Fixes

- Extended the maximum number of instrument actions supported from 64 to 128. This was limiting the use of all actions in some M3xxxA products with more than 64 actions.
- Enhanced the sequence text output to display the instruction including parameter errors, instead of only the instruction label when not all parameters of the instruction are properly set.

## Release 2020 Update 1.1 version information

Build #	1.4.15
Released Date	March 24th, 2021
Operating systems	Microsoft Windows 10 64-bit Pro and Enterprise
Supported Hardware <sup>(1)</sup>	M3100A, M3102A, M3201A, M3202A, M3300A, M3302A, M5302A
File Name	PathWaveTSE_2020Update1.1_winx64.exe

<sup>(1)</sup> For information on the specific product models supported and SW&FW version requirements refer to: [Instrument Software and Firmware Requirements for KS2201A](#)

### New features

- Writing groups of triggers with multiple values synchronously in a single Trigger-Write instruction.
- Added support for new PXIe chassis with integrated high frequency reference clock (slot-19).

### Fixes

- Solved timing skew (~20ns) between M5302A and M3xxxA product families.
- Added support for statement duration property in the text-mode sequence output.
- Improved sequence compilation to allow multiple consecutive Add-Delay statements.
- Improved trigger usage as sync resources to avoid wrong "not enough triggers" error message in some corner cases using multisequence block with triggered-synchronization.
- Improved trigger resources utilization and chassis configuration to solve resync timing issues on some corner cases using 3 or more PXIe chassis depending on the order HVI engine were added into the System-Definition.
- Fixed instability caused when using `Hvi.stop()` in sequences that contain *Wait-for-time* statements.
- Fixed temporary files left behind when using PathWave FPGA .K7z files.

## Release 2020 Update 1 version information

Build #	1.4.7
Released Date	January 6th, 2021
Operating systems	Microsoft Windows 10 64-bit Pro and Enterprise
Supported Hardware <sup>(1)</sup>	M3100A, M3102A, M3201A, M3202A, M3300A, M3302A, M5302A
File Name	PathWaveTSE_2020Update1.0_winx64.exe

<sup>(1)</sup> For information on the specific product models supported and SW&FW version requirements refer to: [Instrument Software and Firmware Requirements for KS2201A](#)

### New features

- Enhanced timing management and added a new Duration property for Sync and Local-flow-control statements.
- .Net API.
- Text-mode sequence export for troubleshooting and debugging.
- New PathWave License Manager 2.3 with support for transportable licenses.
- Improved Python API for consistency and full compliance with PEP-8 naming convention. See breaking changes below.
- Added support for M5302A Digital IO PXIe module.

## Breaking changes

- Python API:

Before	Now
<del>keysight_hvi. Condition. <b>Or()</b></del>	keysight_hvi. Condition. <b>logical_or()</b>
<del>keysight_hvi. Condition. <b>And()</b></del>	keysight_hvi. Condition. <b>logical_and()</b>
<del>keysight_hvi. Condition. <b>Not()</b></del>	keysight_hvi. Condition. <b>logical_not()</b>
<del>keysight_hvi. InstructionSet.trigger_write.sync_mode. <b>IMMEDIATE</b></del>	keysight_hvi. InstructionSet.trigger_write.sync_mode. <b>immediate</b>
<del>keysight_hvi. InstructionSet.trigger_write.sync_mode. <b>SYNC</b></del>	keysight_hvi. InstructionSet.trigger_write.sync_mode. <b>sync</b>
<del>keysight_hvi. InstructionSet.trigger_write.sync_mode. <b>ON</b></del>	keysight_hvi. InstructionSet.trigger_write.sync_mode. <b>on</b>
<del>keysight_hvi. InstructionSet.trigger_write.sync_mode. <b>OFF</b></del>	keysight_hvi. InstructionSet.trigger_write.sync_mode. <b>off</b>
<del>keysight_hvi. RegisterBase. <b>fullname()</b></del>	keysight_hvi. RegisterBase. <b>full_name ()</b>
<del>keysight_hvi. RegisterView. <b>fullname()</b></del>	keysight_hvi. RegisterView. <b>full_name ()</b>
<del>keysight_hvi. InstructionContext. <b>engine_uniqueid()</b></del>	keysight_hvi. Condition. InstructionContext. <b>engine_unique_id()</b>
keysight_hvi. IfBranch. <b>conditional_expression()</b>	keysight_hvi. IfBranch. <b>condition()</b>

## Fixes

- Synchronization issue when using 2+ PXIe chassis fully populated, that could result in 100ns / 200ns skew across modules in different chassis.
- IEvent::Occurred() and Wait() functionality in API that showed wrong behavior and dependency between Occurred() and Wait() calls.

## Release 2020 Update 0.2 version information

Build #	1.0.18
Released Date	October 9th, 2020
Operating systems	Microsoft Windows 10 64-bit Pro and Enterprise
Supported Hardware <sup>(1)</sup>	M3100A, M3102A, M3201A, M3202A, M3300A, M3302A
File Name	PathWaveTSE_2020Update0.2_win_x64.exe

<sup>(1)</sup> For information on the specific product models supported and SW&FW version requirements refer to: [Instrument Software and Firmware Requirements for KS2201A](#)

### Fixes

- Fixed issues in HVI *InstructionSet* definitions:
  - Improved documentation for Instruction parameters.
  - Added missing *id* property in instructions parameters.
  - Added list of possible parameter values for *Value* and *SyncMode* parameter in *TriggerWrite* instruction.
- Fixed issue acquiring PXI triggers in systems with 3 or more chassis (error "*HW trigger PxiTriggerX not acquired*").
- Improved simulated chassis support. Enabled by default the Enhanced-PXI-trigger functionality available on latest FWs for M9018B, M9019A and M9010A chassis. Improved also the User Manual section related to using simulated chassis.

## Release 2020 Update 0.1 version information

Build #	1.0.14
Released Date	September 16th, 2020
Operating systems	Microsoft Windows 10 64-bit Pro and Enterprise
Supported Hardware <sup>(1)</sup>	M3100A, M3102A, M3201A, M3202A, M3300A, M3302A
File Name	PathWaveTSE_2020Update0.1_win_x64.exe

<sup>(1)</sup> For information on the specific product models supported and SW&FW version requirements refer to: [Instrument Software and Firmware Requirements for KS2201A](#)

### Fixes

- Fix problem with licensing that caused trial and possibly other licenses to fail to be recognized correctly. Also updated license installation instructions.
- Moved HVI core library installation location to the common folder and unified it with M3xxxA SW drivers (SD1 software) installation. This requires updating SD1 software to version  $\geq 3.00.95$ .

## Release 2020 version information

Build #	1.0.11
Released Date	August 14th, 2020
Operating systems	Microsoft Windows 10 64-bit Pro and Enterprise
Supported Hardware <sup>(1)</sup>	M3100A, M3102A, M3201A, M3202A, M3300A, M3302A
File Name	PathWaveTestSyncExecutive_2020_shp_win_x64.exe

<sup>(1)</sup> For specific product model supported and SW and FW version requirements check: [Instrument Software and Firmware Requirements for KS2201A](#)

## Features

PathWave Test Sync Executive 2020 is the first release of KS2201A software, it includes:

- Python API for Hard Virtual Instrument (HVI) technology to develop programmatically all phases of HVI technology deployment:
  - System Definition.
  - Sequence programming and compilation.
  - HVI execution.
- Support for up to 6 PXI chassis (actual number of chassis depends on PCIe bus enumeration capabilities which depend on the PC bios, check <https://www.keysight.com/us/en/assets/7018-02925/technical-overviews/5990-7632.pdf> for details)
- Support for M9031A for multi-chassis operation
- New structured programming API for synchronous (global) statements, including:
  - Sync While
  - Sync Multi-Sequence Block
  - Sync Register Sharing (limited to ~4bits given by available PXI trigger resources)
- New programming API for local (single HVI engine and module) statements, including:
  - HVI built-in (or Native) Instructions
    - ActionExecute, TriggerWrite, HVI register AssignAdd/Subtract, FPGA ArrayRead/ArrayWrite/RegisterRead/RegisterWrite



- Product-specific instructions (see product documentation for details)
  - Local If
  - Local while
  - Wait for an event or trigger condition
  - Wait for a Variable time given by a register or add a constant delay
- Support for 32 and 48 bits HVI registers (actual support depends on each product, please check product documentation for details)
  - Integration with Pathwave FPGA
    - load custom bit files (.k7z) and automatically discover user FPGA resources connected to HVI engine (register and memory maps)

## Known Issues or Limitations

- Autocomplete functionality or IntelliSense (in Visual Studio Code) does not work for objects of HVI defined-types when returned from a method. A work around is to explicitly define the type of the object (cast it) to get the autocomplete information. All API information is available in the Python help file installed with PathWave Test Sync Executive 2020.
- The HVI sequence memory in M3xxxA modules is limited to 1024 Instruction-Blocks. The exact number of instruction-blocks required per statement depends on the statement type and timing between statements. Native Instructions take between 1 and 2 Instruction-blocks while Sync and Flow-control statements require several instruction-blocks (between 5 and 10). For typical use cases the total number of statements is between 500 and 1000.



This information is subject to change without notice.

© Keysight Technologies 2020-2023

Edition 2023B\_U0\_00, October, 2023

Printed in USA

KS2201-90006

[www.keysight.com](http://www.keysight.com)