ID	Title
3589	Port Configuration on Framework Side is Incomplete
	Description
	Section 4.1.2 of the Programmer's Guide describes the configuration of the Framework. When calling the method Init() of the Framework, the ports specified by the Framework configuration are instantiated and configured. Furthermore, every port is set to a predefined state also specified by the Framework configuration. Setting a port's state is done by calls to port-specific methods. To set a port's state some of these methods need additional parameters, e.g. MAPort: the method Configure() needs the additional parameter 'forceConfig' to define the behavior of the port if a simulation of a model is already running. • ECUCPort: to set the state 'eONLINE', the method StartOnlineCalibration() with the parameter 'loadingType' must be called. • EESPort: to set the state 'eDOWNLOADED', the method SetErrorConfiguration() with the parameter 'configuration' must be called. Problem: The Programmer's Guide does not describe how to set this additional parameters, if the Framework initializes these ports. Solution: As a product consumer, please contact the Framework vendor to get detailed information about the implemented default behavior that is not specified in the XIL standard yet.
ID	Title
3696	Signal Descriptions: Value at the Boundary between Two Successive Signal Segments is Not Specified
	Description
	The Programmer's Guide introduces the concept of signal descriptions consisting of various signal segments. Problem: The XIL standard does not define the value at the boundary between two successive signal segments. Thus, it is unclear whether the preceding or succeeding segment determines the signal value at the segment transition. Example: A signal description consists of two consecutive const segments with the values 1.0 and 3.0 respectively. The duration of the first const segment is 10.0 [s]. Is the signal value at 10.0 [s] 1.0 or 3.0? Solution: As a product consumer, please contact your Testbench vendor to get detailed information about the implemented be behavior, that is not specified in the XIL standard yet.

ID	Title
3698	Various Errors in the Description and Figures of ECUMPort and ECUCPort in the Programmer's Guide
	Description
	Problem: In various places of section 5.4 "ECUMPort" and section 5.5 "ECUCPort" of the Programmer's Guide, wrong states of the ECUMPort and ECUCPort are used, e.g. the states 'eONLINE' and 'eOFFLINE' for the ECUMPort. Furthermore, in some figures of state diagrams and sequence diagrams and also in the descriptive text wrong or non-existing methods of ECUMPort and ECUCPort are used, e.g. the non-existing constructor ECUMPort.ECUMPort() or the method ECUCPort.Start() instead of ECUCPort.StartOnlineCalibration(). Solution: Refer to the UML model and/or interface definitions of Testbench, ECUMPort and ECUCPort for the correct identifiers of states and the available methods.

ID	Title
3713	Redundant Post Conditions for EESPort Methods
	Description
	Problem: Various methods of the EESPort have the following post conditions: eCOMMON_INVALID_STATE_CHANGE and eEES_INVALID_STATE_FOR_OPERATION (see post conditions in the EESPort in the generic UML model). In case of an invalid operation with respect to the current state, it is not clear, what exception is raised in case of an error. Solution: As a product consumer, you need either consider both post conditions or contact your testbench vendor to get detailed information on the preferred post condition.

ID	Title
3766	Missing Methods and Properties in Tables 49 and 51 for State Transitions of ECUCPort und ECUMPort in the Programmer's Guide
	Description
	In table 49 "ECUMPort states" and table 51 "ECUCPort states" in the Programmer's Guide some methods and properties are missing: Table 49: • Property getConfiguration • Property getDAQClock • Method LoadConfiguration Table 51: • Method CreateSignalGenerator

	• Property getConfiguration • Method LoadConfiguration Solution: Refer to the preconditions of the methods and properties of ECUMPort and ECUCPort documented in the UML model.
ID 3769	Title ECUMPort: Motivation for SetMeasuringVariables
	Description The Programmer's Guide introduces the following note in the Chapter 5.4 “ECUMPort”, section 5.4.1 “Overview”: Note: If the user wants to read variables from the ECUMPort it is necessary that these variables are announced to the port as precondition. The announcement is realized with the method ECUMPort.setMeasuringVariables. Problem: The following sequence diagrams “Read a scalar variable value and examine its properties”, “Read an array variable value and examine its properties“ and “Read a matrix variable value and examine its properties” do not refer to this pre-condition, but only check for the variable’s readability by calling the method: IsReadable(variableName). Furthermore, there is a typo in setMeasuringVariables() and must be SetMeasuringVariables(), since this is not a property but a method. Solution: As a product consumer, please contact your Testbench vendor to get detailed information about the necessity of using ECUMPort.SetMeasuringVariables().
ID 3809	Title MAPortBreakpoint cannot be created due to error in MAPortFactory interface
	Description The factory methode MAPortFactory for the creation of MAPortBreakpoints has the following, erroneous function definition: "CreateMAPortBreakpoint(Watcher, BreakpointAction): int" This function definition does not allow to create instances of MAPortBreakpoint. Solution: CreateMAPortBreakpoint(Watcher, BreakpointAction): MAPortBreakpoint
ID 3810	Title Clear specification of ranges for ErrorObject parameters
	Description The meaningfulness of value ranges of the following parameters shall be checked, particularly whether the value '0' shall be included: - Duration (0) - Frequency - Duty Cycle

	- Resistor (0)
ID 3812	Title When should a "not supported by the EES hardware" exception be thrown?
	Description Chapter 5.6.3.1 "EES HARDWARE LIMITATIONS AND EXTENSIONS" specifies what shall happen if a not implemented function of the hardware is used: "In this case, the EES port returns an error when not implemented functions are used. XIL API compatible test cases can be executed but return an error due to lack of functionality of the underneath hardware ECU Port." Chapter 5.6.1.6. also states "State transitions are only successful, if all pre-conditions are fulfilled and no error occurs during the transition. Otherwise the previous state is not changed. Methods, which trigger a state change, will throw an exception if the state change could not be processed successfully." This shall be further clarified, specifically when this exception should occur. It should also be clarified what state the hardware is in after a "not supported" exception occurred.
ID 3813	Title What happens when an exception occurs during execution of an ErrorSet in eActivated?
	Description The standard only states: "State transitions are only successful, if all pre-conditions are fulfilled and no error occurs during the transition. Otherwise the previous state is not changed. Methods, which trigger a state change, will throw an exception if the state change could not be processed successfully." What happens if an electrical error (e.g. relay broken, etc.) or something else unexpected occurs during execution of an active error set? There is not eError state that the system can go to. A user does not necessarily notice that his EES system is not functioning as expected as this could happen e.g. triggered trough a hardware trigger. When this happens there is no method call activated hence there is no way to throw an exception.
ID 3827	Title Clarification of return value of property BaseError.LoadTypeList for interrupt errors
	Description When an interrupt error happens (e.g. Interrupt, InterruptAtPosition), it is not meaningful to state a LoadType, because the expression 'load throw-off' has no meaning with this error category. Therefore, the answer of BaseError.LoadTypeList-Property shall be specified to avoid diverging interpretation of tool vendors.
ID 3829	Title Clarify time response of method EESPort.Deactivate
	Description

	For the practical use, it is important to know when the FIU is free of electrical errors after a deactivation-call, because the further test execution depends on this. Proposal: Return of method call only after the deactivation of all electrical errors.
ID 3850	Title IEESPort.WaitForTrigger Does Not Have a Numeric Representation of Infinite Timeout
	Description The timeout parameter of WaitForTrigger has a valid range of 0 to infinity, but it is not specified what numeric value should be used to represent infinity.
ID 3852	Title AcquisitionConfiguration Name Must be Unique, But It Can Be Changed at Any Time
	Description The standard describes Error 9004 as "eMEASURING_INVALID_CONFIG_NAME: The name of a configuration must not be an empty string. The name of a configuration must be unique among all configurations in the Acquisition." AcquisitionConfigurations are added to the IAcquisition object by calling SetConfiguration, so the verification for unique names can be performed when the acquisition configurations are assigned to the acquisition, but later their names can be modified by using IAcquisitionConfiguration.SetName method which can result in a violation of the rule above.
ID 3853	Title Error eMEASURING_INVALID_RETRIGGERING (9007) should be thrown by AcquisitionConfiguration.setRetriggering
	Description Table 61. Error Overview from ASAM XIL contains a list of error codes and the methods that should be using them. According to that table eMEASURING_INVALID_RETRIGGERING (9007) is only thrown by RecorderConfiguration.setRetriggering when it should also be thrown by AcquisitionConfiguration.setRetriggering.
ID 3854	Title Implementation of IAcquisition Is Singleton
	Description The standard says that Acquisition, AcquisitionRecorder and Stimulation have to be singleton instances. This requirement has some side effects: 1. No two frameworks can be instantiated at the same time because they would share the same singleton instances. This can be bad because each framework would configure the same instances and that would lead to inconsistency. 2. If a framework is initialized after the previous one was shut down, it would inherit the singleton objects and their configuration. If that framework relies on default config values it will not work as expected, so it needs to reset the configurations.
ID 3856	Title FWSegmentTypes Enum is Missing an Entry for FWDataFileSegment
	Description

	FWSegmentTypes enum contains an entry for all possible segment types, but it doesn't have an entry for FWDataFileSegment.
ID 3857	Title IFWScalarVariableSymbol.CreateScalarVariableSymbolByValue(double value) has the Wrong Parameter Type
	Description There are two factory methods to create a FWScalarVariableSymbol in ASAM.XIL.Interfaces. One of them takes a double as parameter, but it should take a ScalarVariable.
ID 3858	Title SmartAccess.SearchTestbenchLabel Only Returns the First Label with a Matching ID
	Description According to ASAM XIL specification, there can be multiple labels with the same ID as long as they belong to different ports. The standard specifies a helper function to retrieve the testbench label form the TestBenchLabelList that matches the TestBenchLabelRefernce from the MappingTable. That function is SmartAccess.SearchTestbenchLabel and it returns the first label that it finds with a matching ID. Consequently, the returned testbench label is not well defined and this information is incomplete.
ID 3859	Title IFWNoiseSegment.Seed May Have a Wrong Range in ASAM.XIL.Interfaces
	Description IFWNoiseSegment.Seed has the range [-2147483646, +2147483645]. These numbers have an offset of 2 compared to [minint, maxint].
ID 3861	Title New Consistency Rule for not Allowing NO_DIM in a Computation Table
	Description Computation Tables are used for defining how multiplication and division operations of framework variables are handled if physical dimensions are involved. Each computation table has three physical dimensions ids: Factor1QuotientPhysicalDimensionId, Factor2DivisorPhysicalDimensionId, ProductDividendPhysicalDimensionId. The ASAM XIL specification says that "each of these IDs a corresponding physical dimension has to be defined in the mapping files". There is no mention about NO_DIM, which is the standard's constant for no physical dimension, whether it can be or it can not be part of the Computation Table.
ID 3862	Title Framework Configuration Contains No Information About the Tasks to Use When Reading ECUM Port Variables
	Description ECUM Port variables have a list of tasks that they can be measured in. The task to measure a variable is set by calling IECUMPort.SetMeasuringVariables before start of measuring. However, the Framework configuration contains no information about which tasks to be used for measuring variables that are mapped to an ECUM Port.
ID 3864	Title ASAM XIL Specification Needs Clarifications Regarding Which Variables Can be Part of Condition Watcher's Condition
	Description ASAM XIL does not have any clear reference to which variables can be part of a framework condition watcher condition. It can be deduced that the variables need to be scalars because they have to be part of the measured variables and those can only be scalars. However, a scalar variable can be

	mapped to an item of a testbench vector, testbench matrix, testbench curve or testbench map which could also be fine except that Appendix A "Syntax Of Watcher Conditions" claims that array indexing is not allowed in the conditions watcher condition's syntax.
ID 3867	Title ASAM.XIL.Interfaces.Framework.Measuring.IRecorderResultMemoryWriter Seems to be Added by Mistake
	Description ASAM.XIL.Interfaces has ASAM.XIL.Interfaces\Framework\Measuring\IRecorderResultMemoryWriter.cs that defines IRecorderResultMemoryWriter. Looking at the name of the interface one would expect to see a memory writer, but the properties and the description in the file suggest that it is a file writer. Also, there is no factory method for creating instances. The file was probably added accidentally to the project.
ID 3868	Title Contradiction When setRecorderResultWriter Can be Called
	Description According to the ASAM XIL 2.1.0 specification, table 20, Recorder States, the property setRecorderResultWriter can be called in both states eSTOPPED and eSTARTED. According to section 4.4.3.4 Using RecorderResult Readers and Writers: "The writer of the Recorder can only be changed, if the Recorder is stopped." This is a contradiction to table 20.
ID 3871	Title ASAM.XIL.Implementation.XILSupportLibrary:SmartAccess.GetUnitByName Can Throw Error When Creating Variables
	Description UnitId is optional for framework labels in the framework mapping file. If the unit id is not specified in the mapping file, then SmartAccess.GetUnitByName throws an exception when the variable is created because the UnitName is an empty string.
ID 3873	Title Bug in XIL Testbench Exception Implementations and Unification of Testbench and Framework Exception Message Handling
	Description The testbench exceptions as defined in the file "TestbenchPortException_impl.cs" does not consider the fact that the property CodeDescription relies on the property Message of the base class, so that CodeDescription no longer returns the separate CodeDescription, but the message composed of CodeDescription and VendorCodeDescription instead.
ID 3880	Title IPlayer.State has Wrong Type in ASAM.XIL.Interfaces
	Description The type of IPlayerState in ASAM.XIL.Interfaces is StimulationState. It should be PlayerState.
ID 3885	Title Installation of a XIL 2.1 Compliant Implementation Manifest File Causes Errors in a Manifest Reader of XIL API 2.0.x
	Description The installation of an ASAM XIL 2.1 compliant implementation manifest file (.imf) with an ASAM XIL 2.1 compliant server causes errors in a manifest reader of ASAM XIL version 2.0.x. Therefore, ASAM XIL 2.0.x compatible servers can not be used afterwards.

	The root cause is the incompatibility of the ASAM XIL 2.0.x manifest readers with ASAM XIL 2.1 compliant manifest files containing version 2.1 in the XML namespace of .imf files. The manifest readers assume that all .imf files contained in folder %ProgramData%\ASAM\XIL\Implementation are XIL 2.0.x-compliant and use version 2.0 in their namespaces.
ID	Title
3886	SignalDescriptionParameter: Missing Stereotype <getter> at getDefault-Value in Generic UML Model
	Description
	In the generic UML model of ASAM XIL 2.1 the stereotype <getter> is miss-ing at the operation getDefaultValue in class Testbench.Common.Sig-nal.SignalDescriptionParameter. This leads to a method "getDefaultValue" in the C# interfaces instead of a read-only property "DefaultValue".
ID	Title
3925	Clarify Reference Time for DurationWatcher Breakpoint
	Description
	In ASAM XIL 2.1 the reference time of MAPortBreakpoint with Duration-Watcher is ambiguous. It is not clear when a breakpoint with Duration-Watcher shall pause or stop the simulation. The duration may refer to the simulation timestamp when the breakpoint was assigned to the MAPort, or to the timestamp of the most recent status change to eSIMULATION_RUN-NING.
ID	Title
3928	SignalGeneratorWriter in Wrong Package
	Description
	In ASAM XIL 2.1, SignalGeneratorWriter has been moved to package ASAM.XIL.Interfaces.Testbench.Common.Script by accident. This breaks binary and source code compatibility with existing clients. For instance the ASAM XIL test-suite cannot be compiled against ASAM XIL 2.1 because of that bug.
ID	Title
3957	Validity Scope of Instances
	Description
	The ASAM XIL Programmer's Guide has requirements that sometimes can't be achieved unless the implementation of an interface is known. However, with some exceptions, the specification has no limitations about where an instance can be used. For now, instances can be used anywhere as ex-pected by the parameter that describes the type of the interface.
ID	Title
3966	Remove XYValue from ValueContainer Package
	Description
	In contrast to MapValue, the interface CurveValue is not directly derived from BaseValue. Instead there is an intermediate interface XYValue. Neither the semantics nor the purpose of XYValue are clearly described. Further-more, the implementation of XYValue and CurveValue in the XILSupportLi-brary are identical. The suggestion is to remove the XYValue with the next major release.
ID	Title
	Description

About This Document

This document lists known issues for the standard and version as identified in the document header. Issues in the context of ASAM standards have one of the following characteristics:

- Error: unintended or wrong content.
- Contradiction: inconsistent or contradictory content.
- Specification gap: missing content required for a functional system and for complete understanding.
- Lack of clarity: Unclear, vague or ambiguous description, which leads to misunderstandings and misinterpretations.

The issue may exist in the base standard, in associate standards, schema files, interface definition files, model files, examples or any other supplements of the standard.

For each issue, the table contains an ID, title and description.

ID: Unique identification number assigned by the ASAM change request system.

Title: Summary of the issue description in headline style

Description: Identifies the parts of the standard that are affected by the issue, provides a reason why this is considered as an issue and allows the reader to understand the technical implications of the issue. Optionally, the description includes a resolution proposal and a proposed workaround for the issue.

Issue are resolved in the release of a new version of a standard. Please regularly check ASAM's web page and news publications to stay informed about new versions. If an issue has been resolved in a new version, then it is not listed in the List of Known Issues document for this version any longer.

The List of Known Issues document for former versions of the same standard will be frozen and will not be further maintained. ASAM advises all users of its standards to always use the latest version of its standards.