IBIS Quality Task Group Report – IBISCHK8 Syntax Parser

Randy Wolff

Siemens EDA

Chair, IBIS Quality Task Group

European Hybrid IBIS Summit with IEEE SPI 2025 Gaeta, Italy May 14, 2025



An SAE Industry Technologies Consortia Program

IBIS Quality Task Group

- Manages IBISCHK and TSCHK parser testing and development
 - Parser contract development
 - Bug report reviews and classifications
- Develops the IBIS Quality Specification
 - Current version is 3.0
 - Available at: https://ibis.org/quality-ver3.0/
- To get involved:
 - Teleconference meetings held as-needed on Tuesdays as 9:00am US Pacific Time
 - Website: https://ibis.org/quality_wip/
 - Email discussions: <u>https://www.freelists.org/list/ibis-quality</u>
- Chair: Randy Wolff, Siemens EDA

Update on IBISCHK8 Syntax Parser Development

- Parser contract developed in November/December 2024
 - Quote from developer estimated ~44 days and \$17,500
 - Includes test suite preparation and documentation
- Reviews of BIRD223.1 and BIRD226 revealed some syntax parsing issues
 - Questions sent to Editorial Task Group for BIRD clarifications
 - Many addressed in latest IBIS 8.0 specification draft
- Discussions have resumed in ATM task group on Power Integrity Modeling
 - IBIS 8.0 editorial work is on-hold for now
 - IBISCHK8 parser work is on-hold for now

IBISCHK8 Parser Coverage

Current official BIRD content – approved June 21, 2024

BIRD ID	BIRD Title	Approval Date	Note	
223.1	Add support for SPIM in IBIS	November 17, 2023	Power Aware	
224	New AMI Reserved Parameters for Ts4file port order	June 23, 2023	AMI	
225	Clarification for bus_label rules	September 15, 2023	Power Aware	
226	PSIJ Sensitivity	December 8, 2023	Power Aware	
227	AMI Ignore Block Feature	January 5, 2024	AMI	
228	Pin Name Field Extension	January 26, 2024	General	
229.1	AMI Test Data Support	April 19, 2024	AMI, Quality	
230.1	Adding a Definitions Section to IBIS	May 31, 2024	General	
231	Clarifications on AMI Block Concepts	May 31, 2024	AMI	

Current BUGs filed against IBISCHK7.2.1 (9 BUGs, some Severe and Fatal Severity)

ID#	Title	Requester	Date Submitted	Severity	Priority	Status
254	AMI Reserved Parameter Repeater_Type erroneously reported as missing	Arpad Muranyi, Siemens EDA	April 24, 2025	[MODERATE]	[LOW]	[OPEN]
253	Non-comments allowed after [End Interconnect Model Set] keyword	Justin Butterfield, Chinmaya Patra, Bibhu Panda; Micron Technology	March 31, 2025	[MODERATE]	[MEDIUM]	[OPEN]
252	Incorrect BCI Interpretation Invalidates All Tx Models	Michael Mirmak, Intel Corporation	March 17, 2025	[SEVERE]	[HIGH]	[OPEN]
251	Missing Repeater_Type not seen through Model Selector	Arpad Muranyi, Siemens EDA	November 15, 2024	SEVERE	HIGH	OPEN
250	Incorrect [Diff Pin] caution report	Michael Mirmak, Intel Corp.	July 1, 2024	ANNOYING	LOW	OPEN
249	Memory leaks caused by algmod.c, ami.c and cmn.c	Michael Mirmak, Intel Corporation	March 28, 2024	ANNOYING	LOW	OPEN
248	Crash Caused by iassert macro and abort()	Michael Mirmak, Intel Corporation	March 19, 2024	FATAL	MEDIUM	OPEN
247	Changing BOOL from Type Enum to Int	Michael Mirmak, Intel Corporation	March 19, 2024	ENHANCEMENT	LOW	OPEN
246	Clarification on Slash Characters and Parser Usage	Michael Mirmak, Intel Corporation	December 22, 2023	MODERATE	MEDIUM	OPEN

IBISCHK8 Parser Requirements (the easy stuff)

- BIRD224 "New AMI Reserved Parameters for Ts4file port order"
 - Support two new AMI Reserved Parameters "Tx_Port_Order" and "Rx_Port_Order"
- BIRD225 "Clarification for bus_label rules"
 - No parser changes needed
- BIRD227 "AMI Ignore Block Feature"
 - Support one new AMI Reserved Parameter "Adaptation_Valid"
- BIRD228 "Pin Name Field Extension"
 - Pin Name field extended from 5 to 20 characters, affecting character limit checks on 10 keywords
- BIRD230 "Adding a Definitions Section to IBIS"
 - No parser changes needed
- BIRD231 "Clarifications on AMI Block Concepts"
 - No parser changes needed

IBISCHK8 Parser – BIRD223.1

- BIRD223.1 "Add support for SPIM in IBIS"
 - Adds keyword pair [Device SPIM Group]/ [End Device SPIM Group] to a .ibs file scoped under [Component].
 - Creates new file format with extension "spim".
 - Individual .spim files will require the -spim flag.
 - If directly checking a .spim file, [Pin] information found only in the .ibs file will be missing, so some checks that would cross check to the [Pin] list cannot be performed.

IBISCHK8 Parser – BIRD223.1

- SPIM file parsing
 - Checking of Header section ([IBIS Ver], [File Name], etc.) and [End] keywords follows same rules as .ibs,.pkg. .ebd, .emd, .ems, and .ims files
- SPIM file tree structure shows scale of syntax checking required:



IBISCHK8 Parser – BIRD223.1

- SPIM file parsing (going beyond simple checks)
 - When checking is initiated from a .ibs file, additional cross checking is done:
 - [Device SPIM] will check for matching of [Device SPIM Group] entry
 - [SPIM Rail] will check for matching signal_name or bus_label names from the [Pin] keyword, the [Pin Mapping] keyword, or the [Bus Label] keyword
 - [SPIM Pin Cluster] will check for matching pin names from the [Pin] keyword
 - [SPIM Port List] can extend checks to referenced Touchstone files for specific conditions such as containing at least 3 ports
 - [SPIM Rnetwork File]
 - Check that .subckt name exists in the referenced file
 - Additional rules are under development for checking number of subcircuit terminals and terminal names
 - Additional file parsing would require IBIS-ISS syntax parser

IBISCHK8 Parser – BIRD226

- BIRD226 "PSIJ Sensitivity"
 - Adds 4 new keyword pairs scoped under [Component]
 - Multiple [PSIJ Sensitivity] sections are allowed under a [Component] but check for duplicate [PSIJ Sensitivity] names
 - In [PSIJ Sensitivity Rail]:
 - Require string argument format "PSR_of_<signal_name>".
 - Check that <signal_name> exists in the [Pin] list (must be POWER/GND model type)
 - Check that one signal_name is associated with a POWER pin and one signal_name is associated with a GND pin
 - For [PSIJ Sensitivity Signal]:
 - Check that the model_name listed exists within the [Pin] list as a [Model] or a [Model Selector] name, but not POWER, GND, or NC
 - Check that there are 100 or less frequency points, arranged in increasing order of frequency
 - For [PSIJ Voltage List]:
 - Follow consistency checking used for [Voltage List] in .emd files

IBISCHK8 Parser – BIRD229.1

- BIRD229.1 "AMI Test Data Support"
 - Introduces new keyword [AMI Test Configuration] scoped under [Algorithmic Model].
 - Includes 11 subparameters: Type, Direction, AMI_input_parameters_file, Input_IR_file, Input_waveform_file, Golden_IR_file, Golden_waveform_file, Clock_input_file, Clock_output_file, AMI_output_parameters_file, and Executable_index.
 - Tree structure view:



IBISCHK8 Parser – BIRD229.1

- BIRD229.1 "AMI Test Data Support"
 - Significant syntax checking goes beyond subparameter parsing
 - Included files to be parsed to ensure they follow content rules such as:
 - Verifying parameter tree structures (like checking of a .ami file),
 - Correct number of columns of data
 - Minimum number of rows of data
 - Cross checking for same number of rows of data for specific files
 - Specific data formats such as floating-point numbers, etc.
 - AMI_input_parameters_file has additional 8 rules to check
 - AMI_output_parameters_file has additional 7 rules to check
 - Checking of root name as requested in BUG227 for the ibischk7 parser (not implemented in IBISCHK7.2.1) is enabled by this keyword, allowing IBISHCK8 to exercise AMI DLLs to output a root name for verification

IBISCHK8 Parser – BUG227 Root Name Checks

- For any instance of [AMI Test Configuration], the parser should use the associated files as inputs to the AMI_Init and/or AMI_GetWave functions.
- The root name found in the AMI_parameters_out string can then be compared to the root name in the .ami file.
- An [AMI Test Configuration] of Type Statistical will have inputs to test the AMI_Init function.
- An [AMI Test Configuration] of Type Time_domain will have inputs to test the AMI_Init and AMI_GetWave functions.
- A single call to AMI_GetWave is sufficient for generating a parsable AMI_parameters_out string. This call will pass into the AMI_GetWave function the first set of data of size wave_size from the Input_waveform_file.
- Parser contract defines 5 valid [AMI Test Configuration] setups

IBISCHK8 Parser – Next Steps

- Quality task group will wait for IBIS 8.0 discussions to complete, then:
 - Update parser contract with any new/modified BIRDs
 - Update parser contract with any new BUG reports
 - Get new quote from parser developer
 - Target parser completion to coincide with IBIS 8.0 specification final approval vote

[Thank You]

