SiliconSoftware
speed up your vision

VisualApplets

Version 3.1

Release Notes

VisualApplets Version 3.1

Update Recommended

Due to the extensive number of enhancements available with this new version, we generally recommend to update your VisualApplets installation to VisualApplets version 3.1.

New Features

New Hardware Platform mE5 marathon VCLx

- Full support of the new frame grabber board microEnable 5 marathon VCLx.
    - mE5 marathon VCLx is a fully programmable Camera Link frame grabber fit for especially complex image processing as well as deep learning applications based on CNN (Convolutional Neuronal Networks).
    - marathon VCLx is equipped with an extremely powerful FPGA, thus providing extended FPGA ressources.
    - marathon VCLx provides 2 camera ports supporting the Camera Link standard in version 2.0 (Base, Medium, Full, and Full Deca 80 bit configuration).

Windows 10 Support

- VisualApplets now fully supports the operating system Microsoft Windows 10.

TCL Scripting – New Feature in VisualApplets Expert

The VisualApplets extension **VA Expert** now includes TCL scripting. Whole designs, as well as design parametrization and any simulation procedures can be described now in TCL script. A huge set of VisualApplets specific commands is available.

This allows for

- **Automatizing the creation of designs**: Use scripts to assemble components of your design controlled by parameters. Set up libraries of processing pipelines and create bunches of different designs automatically.
- **Automatizing the simulation of designs**: Use scripts for defining extensive simulation runs. A TCL script may execute series of simulation runs over thousands of images stored somewhere in your file system with the option to vary design parameterization between the runs and adding means for automatically analyzing the simulation results.

Console: VisualApplets now offers a TCL console from where you can work interactively with the GUI by entering commands. From here you can call scripts or develop new scripts by pasting commands and trying if they work as expected. In addition, many actions you carry out via the VisualApplets GUI are logged in the console as corresponding Tcl commands so that you can check how your command may look like in TCL. The new feature is supplemented with an extensive command reference and a set of TCL scripting examples.

Separate Licenses for Debugging Library and Parameters Library

VisualApplets Release 3.1 offers an extended licensing scheme. Two libraries that are integral parts of the *VisualApplets Expert* extension can be purchased now also as separate modules by users who are not interested in the full *VisualApplets Expert* extension. Upon purchase, an according license is issued for the respective module by Silicon Software. All other aspects of the VisualApplets licensing scheme remain untouched. The handling of the new licenses corresponds to the general handling of VisualApplets licenses.

The following features can now be purchased separately:

- **Debugging Module**
  The *Debugging Module* contains the *Debugging* library. The *Debugging* library offers a variety of operators with enhanced debugging capabilities. They can be used for debugging *VisualApplet Designs* in the real runtime environment as well as for debugging *custom operators* which are still under development.

- **Parameters Translation Module**
  The *Parameters Translation Module* contains the *Parameters* library. Library *Parameters* allows to set paths to specific parameters which may be deeply embedded in hierarchical

structures. This way, even parameters in protected hierarchical boxes (or in protected hierarchical boxes that are nested in other protected hierarchical boxes) can be set and reset during design time and runtime, since their parametrization interface is lifted up in the hierarchy to a hierarchical level that is not access restricted. The translation operators of the library in addition allow translated access (read and write) to parameters of one or several other operators. For calculation, formulas can be defined according to the GenICam API 2.0 standard.

Operators

- **New Operators**
  - In library ***Parameters***, three new operators are available:
    - IntVariable
    - FloatVariable
    - EnumVariable

These operators allow to define pure software parameters which may be referenced by other operators.

Extended Examples

- **New Examples**
  - The set of VisualApplets design examples has been further extended and now also contains processing examples for
    - Normalized Cross Correlation (Processing\ObjectFeatures),
    - Exposure Fusion (Processing\HDR_ImageComposition)
    - Trigger Statistics (Processing\DebuggingAndTest),
    - Distortion Correction (Processing\Geometry\GeometricTransformation), and
    - Tap Sorting with BRAM (Processing\Geometry\TapGeometrySorting). (7320)
- **Sim Modules Contain Images**
  - All of the processing examples that contain simulation modules now have the images for simulation already available in the simulation source modules. You can simply open the examples and immediately run simulations without loading simulation images into the simlation source modules. (7018)

Version Check Feature

You can check on newer VisualApplets versions directly from the GUI via mouse click, or configure VisualApplets to automatically check for newer runtime versions at each program start.
Requirements: Internet connection and an appropriate firewall exception.

Changes and Bug Fixes

Operators

- **Three operators no longer availabe for mE5 and eVA platforms**
  - The following three operators:
    - ImageBuffer_JPEG_Gray
    - ImageBufferSpatial
    - ImageSequence

  are no longer available in designs for microEnable 5 or eVA (embedded VisualApplets) platforms. They can still be used in designs for target platforms of the microEnable 4 series (including PixelPlant).

- **Parameters Library**
  - **Enhanced formula editor:** The formula editor of the Parameters library has been enhanced. It offers now syntax highlighting and autocompletion of operator paths and variable names.
  - In the operators of the **parameters** library, entering a slash at the end of the value for parameter **DisplayHierarchy** during design time is inhibited. This avoids problems during runtime, as during runtime, a slash at the end of parameter **DisplayHierarchy** makes the parameter inaccessible (read only), even when the parameter is a dynamic parameter (read/write). (8262)
  - Various improvements of error handling.
  - **Changing parameter type via display parameter:** Fixed changing the parameter type between DYNAMIC and STATIC from a display parameter of a ***HierarchicalBox*** (parameter generated by a ***Parameters*** operator when ***DisplayName*** has been specified). Before, this operation could lead to side effects leading to errors in the runtime environment. (8232)

- **MULT**
  - Operator **MULT** now supports asymmetric arithmetic types on its inputs, i.e., SIGNED and UNSIGNED may be mixed up so that you can multiply an unsigned multiplier with a signed multiplicand or a signed multiplier with an unsigned multiplicand (as well as a signed multiplier with a signed multiplicand or an unsigned multiplier with an unsigned multiplicand). (2729)
- **RamLUT**
  - Operator **RamLUT** supports kernels on the output. (4091)
- **StreamAnalyzer**
  - Operator **StreamAnalyzer** now also supports LINE1D and PIXEL0D formats. (7120)
- **ImageInjector**
  - Incoming links with a bit width up to 64 bit are supported. (In earlier versions, when bit width on the the incoming link of ImageInjector was set to more than 32 bit, the operator didn't insert correct image data. Fixed.) (7366)
  - The operator ***ImageInjector*** will not create additional phantom frames anymore. (In earlier versions, under special cicumstances it was possible that a single injected frame caused several frames. Fixed.) (7247)
  - Operator ImageInjector works correctly now. (In version 3.0.6, there occurred problems loading an image. Fixed.) (8201)
  - Parameter mode has a new mode ***Inject***. In mode ***Inject***, the operator immediately blocks the input and you can inject an image from file. (6877)
- **MediumRgbCamera**
  - Order of color components matches now Camera Link Specification 2.1. (In earlier versions, green and blue were swapped in modi Tap3x10bit and Tap3x12bit. Fixed.) (7476)
- **Blob_Analysis_1D / Blob_Analysis_2D**
  - Operator works as expected also when the line marker is disabled. (In earlier versions, when parameter ***line_marker*** was set to ***not_used***, an error message occurred during design rules check (DRC 2). Fixed.) (7595)

- **IsLastPixel**
  - Operator IsLastPixel now creates correct images also if its output ports are connected to asynchroneous sinks, i.e., also if its output ports receive differing data flow signals (for example, if only one of the output ports was stopped). (In earlier versions, all output ports of IsLastPixel needed to be connected to the same sink (M operator) to make the operator work stable. Fixed.) (7731)
  - The operator link IsEmptyO works now as expected. (In earlier versions, under specific circumstances a data loss could occur on this link. Fixed.) (7729)
- **Synthesis of Designs Containing Multiple SQRT Operators**
  - Synthesis of designs containing multiple instances of the SQRT operator works now as expected. (In earlier versions, synthesis was not successful when multiple SQRT operator instances in the same design used differing bit depth configurations. Fixed.) (7739)
- **RxImageLink – SDK Generation**
  - SDK code generated out of a design containing operator RxImageLink is correct now. (In earlier versions, the generated code contained invalid syntax for the parameter definition of RxImageLink which led to compiler errors. Fixed.) (7757)
- **Insert Port Above/Below for all Operators with Variable Number of Ports**
  - All operators with a variable number of ports now offer a context menu “Insert Port Above/Below”. (Only Exception: Operator SignalToEvent) (7786)
- **EnumParamTranslator**
  - Enumeration names in operator EnumParamTranslator may contain now ','. (In earlier versions, during runtime it came to problems when an enumeration name contained a comma. Fixed.) (7860)

    Attention! Using comma in enumeration names is not allowed when generated GenICam API code shall be used for eVA platforms (platforms working with embedded VisualApplets).
- **SelectBitField allows Color Input**
  - Operator SelectBitField (Library Base) now also allows color input. (8076)

- **Color support of RND, MergePixel, SelectBitField, ShiftLeft, ShiftRight / New Round (RND) implementation**
  - The following operators now also support color formats. Each color component is processed separately. (6973)
  - RND has a new Implementation which requires only about the half of the LUTs that have been required for rounding operations before. The behavior of the operator is identical to previous versions. (6973)
  - Affected operators:
    - RND (library Arithmetics)
    - MergePixel (library Base)
    - SelectBitField (library Base)
    - ShiftLeft (library Arithmetics)
    - ShiftRight (library Arithmetics)
- **Operators that can be used to modify the maximum image dimensions on a Link**
  - These operators now verify that the new image dimensions do not exceed the absolute maximum image dimensions (maximum image width: 2^31-1, maximum image height: 2^31-1). (4764)
  - Link parameter / operator parameter collisions leading to an MaxImgWidth overflow are detected now and marked red in the VisualApplets GUI to inform the user instantly. (4764)
  - Affected operators:
    - CastParallel (library Base)
    - PARALLELdn (library Base)
    - PARALLELup (library Base)
    - SampleUp (library Base)
    - PackbitsRLE (library Prototype)
    - InsertPixel (library Synchronization)

GUI

- **VisualApplets Base License contains FPGA Resource Estimation on Module Level**
    - The Feature *FPGA Resource Estimation on Module Level* is now available to all users of VisualApplets. (In earlier versions, it was available only to users of the VisualApplets *Expert* extension. Changed.)
- **New Settings Option: Number of Undo-Steps**
    - In menu Settings -> System Settings -> Design, you can define the number of Undo steps for working with the VisualApplets GUI.
    - If you set the number of Undo steps to 0, option Undo is de-activated.
    - De-activation of Undo leads in complex designs to an enhanced GUI performance. However, if Undo is de-activated, a design cannot be restored after a crash. (7427)
- **Highlighting of selected elements *in Module Info* window**
    - The selected elements in the currently active design window are highlighted in the *Module Info* window. (In earlier versions, this worked only for selections in the Process0 window. Fixed.) (7356)
- **Browsing the Design while Dialogs are open**
    - You can browse (but not change) your design now while the following dialogs are open:
        - Resources
        - FPGA Resource Usage
        - Bandwidth Analysis
        - Simulation
        - Build Hardware Applet
          (7124)
- **Zooming the Help Window via Strg + Mouse Wheel**
    - You can zoom in and out the content of the help window via Strg + mouse wheel. (in earlier versions, only the headings of the help where zoomed. Fixed.) (6978)
- **Enhanced Display of Changed Link Parameters**
    - After the parameters of a link have been changed, the changed link is highlighed in green. Closing the *Link Properties* dialog doesn't change the highlighting. The highlighting disappears as soon as you click on an empty area of the design window.
- **Enhanced Editing of Formulas**
    - The editor for write actions, read actions and range equations in operator instances of library *Parameters* now supports syntax highlighting and auto-completion of module paths and variables.

Simulation

- **Pathes to Simulation Images**
  - New option for saving paths for images in simulation sources relative to a given folder. In menu **Settings/System Settings** in category ***'Design'*** after ***'Paths for Simulation Source Images'*** it can be specified whether the paths shall be absolute, relative to design location, relative to the VisualApplets installation directory or relative to the default simulation image path specified in category ***Simulation***.
- **Simulation Settings**
  - The simulation settings (menu Settings-> System Settings -> Simulation) for the I/O Image Library are set per default to TIFF compression = LZW. This reduces the size of the TIFF images. (7693)
- **Deleting Simulation Probe Modules in H Boxes**
  - Deleting simulation probe modules located at links connected to H box input nodes now works as expected. (In earlier versions, deleting such simulation modules could lead to errors or program crash. Fixed.) (7795)
- **Simulation Probe Viewer**
  - Simulation results are displayed properly now also for image sequences containing images with multiple kernel components and varying image width. (In earlier versions, simulation results from image sequences that had multiple kernel components and varying image width where not displayed correctly in the Viewer. However, this was a mere display bug, simulation itself was not affected. Fixed.) (7705)
- **Simulation of 64-bit Images**
  - Simulation now handles source images with 64-bit components as expected. (In earlier versions, under specific circumstances, after simulating 64-bit images the simulation probe only contained 0 values. Fixed.) (7862)

MCF Files

- Saving the MCF file now also works when there is no file name defined in operator ***CoefficientBuffer***. (In earlier versions, writing the MCF file stopped as soon as there was an empty string detected where a file name was expected in an instance of ***CoefficientBuffer***. This led to incomplete MCF files. Fixed.) (7939)

- MCF files generated for a design containing operator ***CLHSDualCamera*** (VF2) by microDiplay can be loaded in the runtime using functions `Fg_loadConfig` / `Fg_InitConfig` or in microDisplay itself. (In earlier versions, under rare circumstances loading such a file during runtime was not possible. Fixed.) (6158)

Created SDK Code

- **DMA channel status GRAB_ACTIVE**
  - The DMA channel status GRAB_ACTIVE can be retrieved using Fg_getStatus/Ex. It is 1 only when a DMA channel was successfully started and has not yet been stopped again. In all other cases it is 0. (In the Visual Applets version prior to 3.1 this was not the case. Fixed.) (6865)

Other Features

- **Build**
  - Build flow works correctly also when LED and/or GPO operators are used in designs for microEnable 5 platforms. (In earlier versions, using LED and/or GPO operators in designs for microEnable 5 platforms led in some cases to build flow errors (step *Optimize Design*). Fixed.) (8427)
- **Error Message System**
  - The set of error messages for the Design Rules Checks (DRC1 and DRC2) has been enhanced. (8086)
  - When VisualApplets during simulation cannot write image data to hard disc because the hard disk is full, an according error message occurs that provides the path to the intended saving location. (In earlier versions, a message “File access denied” was displayed. Fixed.(7919)
- **Unlimited Number of Design Elements**
  - Internal limit for the number of allowed design elements has been removed. (In earlier versions, this limit led in extremely complex designs to the following CHDL error message during Design Rule Check 2:

    ```
    Error: -1 chdl2.cxx:2677:computeOMLatency-Range error
    ```

    Fixed. (8084)

- **TCL Export/Import**
  - LUT values are saved very efficiently now so that saving to Tcl and loading a Tcl file works much faster. (7790)
  - Floating-point values are now saved to Tcl script absolute precisely.
- **GenAPI XML Generation**
  - The generation of GenAPI XML content for embedded VisualApplets designs has been improved concerning the complicance with 3rd party tools. For embedded VisualApplets platforms new properties may be defined with allow for generating default <Visibility> and <ImposedAccessMode> entries for features which don't specify these tags. (7024)
  - Generating GenAPI XML content for formulas in operators of library *Parameters* was fixed. Before a pure assignment equation for the first of several target write equations could lead to incomplete execution of these equations at runtime. (8105)
- **Unicode Characters in Directory Names supported**
  - For naming directories in paths, unicode characters can be used. However, for naming *.va files, only ASCII characters can be used; this means that, e.g., Asian, Cyrillic, Greek, or Arabic characters are not supported for file names.

Documentation

- ***In Tutorial and Examples***, all sections and subsections of the documentation are contained. (In the last version, in section *Processing Examples*, the subsections *Advanced*, *Binarization*, Blob *Analysis*, *Color*, and *Debugging and Test* were missing. Fixed.) (7327)

VisualApplets Version 3.0.6

New Features

- **New Hardware Platform marathon VCX-QP Supported**
  Designs for the completely new Silicon Software frame grabber board **mE5 marathon VCX-QP** can be designed and built.

- **CXP Camera Operators Allow to Define Incoming Pixel Format**
  The three CXP camera operators CXPSingleCamera, CXPDualCamera, and CXPQuadCamera now allow to define the pixel format of the image data that are coming in from the camera. Supported are all Mono, RGB, and Bayer pixel formats supported by the CXP specification. For all other CXP pixel formats, the pixel format setting “RAW” can be used. The three camera operators are available for all Silicon Software CXP frame grabbers (mE5 marathon VCX-QP, mE5 ironman VQ8-CXP6D, and / mE5 ironman VQ8-CXP6B).

- **Improved Protection of Changes in User Library Instances**
  Instances of user library elements that have been modified by the user are marked now with a special icon. In addition, when an update or quick update from the user library is started on such an instance, an according message occurs that informs the user that all changes that have been made to this instance will be lost if the update is carried out. This way, adapted user library instances cannot be overwritten unintentionally anymore. (4906)

- **Applets for marathon VCL and LightBridge VCL: PoCL Support per Default Deactivated**
  Applets you create with VisualApplets 3.0.6 (or higher) for marathon VCL and LightBridge VCL boards: The automatic PoCL detection feature will be disabled on the frame grabber board running the applet. PoCL support needs to be enabled by the user via microDiagnostics[1] (menu Tools -> Board Settings).

[1] microDisplay comes as part of the Silicon Software runtime software installation. The option for enabling/disabling PoCL support is implemented in microDisplay since runtime version 5.5.1 (and higher).

- **Tcl Export and Import Option**

  VisualApplets 3.0.6 allows to export VisualApplets designs as human readable Tcl script code (*.tcl). This new feature is intended for revision control and for comparing different versions automatically (by creating "diffs"). Earlier exported Tcl script code can be imported into VisualApplets. The Tcl scripts are not intended for being used as the primary file format for saving designs. The generated Tcl scripts do not cover all style information and don’t contain the design structure of instantiated user library elements.

- **IstFirstPixel and IsLast Pixel – Two new Operators in the Synchronization Library**

  Library Synchronization has two new operators:
  - **IsFirstPixel**: Operator IsFirstPixel marks the first pixel in a line (in line mode) or in a frame (in frame mode). The operator outputs a 1 on its output port IsFirstO for each first pixel of a line/frame.
  - **IsLastPixel**: Operator IsLastPixel marks the last pixel of a line (in line mode) / of a frame (in frame mode). The operator can also be used to mark empty lines (in line mode) or empty frames (in frame mode).

- **Examples**

  The set of design examples has been extended:
  - **Print inspection:** Two new examples for print inspection have been added that both, though using different methods, allow object detection with identifying defects and correcting the position of detected objects within an image.
  - **Triggering:** The set of design examples using triggers has been extended. Furthermore, the examples are now available in hardware platform specific variants.
  - **Basic Acquisition:** The set of basic acquisition examples has been extended. Furthermore, the examples are now available in hardware platform specific variants.

Changes and Bug Fixes

Operator TrgBoxLine

Operator TrgBoxLine (library Prototype):

- The value range for parameter ImgTrgDelay has been extended to {0;65535} lines. (6848)
- The operator's output port Exsync works correctly now. (6608)

Memory Operators allow Kernel Size >1

The operators LineMemory, LineMemoryRandomRd, FrameMemory, and FrameMemoryRandomRead (library *Memory*) now allow a kernel size > 1. (2369)

PixelNeighbours1xM Operator

Operator PixelNeighbours1xM now supports variable line lengths also during simulation. (In earlier versions, variable line lengths were only supported in hardware.) (7170)

Camera Link: Improved stability after invalid pixel signals from camera

When the frame grabber received invalid pixels from the camera before the acquisition was started, the first valid line/frame sent by the camera after start of acquisition was discarded (due to a filter system implemented to discard invalid signals) even if the acquisition was started via trigger. Fixed. The filter system has been improved so that now after start of acquisition via trigger the first line/frame sent by the camera is processed also in cases when invalid pixels have been sent by the camera before start of acquisition. The fix has been implemented into all mE5 marathon, mE 5 ironman, and LightBridge Camera Link frame grabbers, and into microEnable IV AD4 CL/PoCL and microEnable IV VD4 CL/PoCL.

PoCL Support Improved

The PoCL support that is automatically implemented in Camera Link Applets for marathon and LightBridge frame grabbers has been improved:

- The PoCL detect mechanism has been further enhanced.
- The support of PoCL cameras that are changing their clock frequency has been improved: Instable clock signals sent by the camera (during reconfiguration of clock frequency in the camera) are tolerated by the applets for a longer period of time.

Instances of User Library Elements: Improved Handling of Changes

Instances of user library elements that have been modified by the user are marked now with a special icon. In addition, when an update or quick update from the user library is started on such an instance, an according message occurs that informs the user that all changes that have been made to this instance will be lost if the update is carried out. This way, changed user library instances cannot be overwritten unintentionally anymore.

Parameters Library

VisualApplets detects when due to wrong parametrization of the parameters "DisplayHierarchy" and "Display Name" a reference/translation operator module tries to add an operator parameter to a process of a design (which is not allowed). In earlier versions, this could lead to a crash of VisualApplets, or to an applet crash during runtime. Fixed. VisualApplets now informs via an error message how to parameterize the reference/translation operator correctly.[2] (7053)

Fix of Blob Operators

In VisualApplets version 3.0.4, Operators Blob_Analysis_1D and Blob_Analysis_2D caused a dead lock when being blocked. Fixed. (6702)

Fix of DIV Operator

The timing closure of the DIV operator (library Arithmetics) has been improved. (In earlier versions, under specific circumstances it happened that the DIV operator caused timing errors during build. Fixed.)(6900)

RamLUT Operator: New Error Code for Initialization File

The RamLUT operator now returns error code FG_INVALID_FILESIZE if the size of its initialization file is not correct. (In earlier versions, the error code FG_ERR_RANGE_ERROR was output in such

[2] If the reference/translation operator is located on the highest hierarchical level of the design (process level) and "DisplayHierarchy" is empty, "DisplayName" must be empty, too, because the process itself is not a hierarchical box and cannot get parameters.
Now, VisualApplets detects such situations and throws the according error message "DisplayName must be empty when DisplayHierarchy refers to process Level". (In earlier versions, it was possible to define a parameter under "Display Name" also when "Display Hierarchy" pointed to the process level (i.e., when the reference/translation operator was located on the highest hierarchical level of the design and "Display Hierarchy" was empty). This could lead to crashes of VisualApplets, or to applet crashes during runtime. Fixed.)

situations.) The new error code FG_INVALID_FILESIZE is available in conjunction with Silicon Software runtime software version 5.4.4.1 or higher. (6422)

Fix of Operator LineMemory

Operator LineMemory has been corrected. In earlier versions, under very rare circumstances there was the possibility of data corruption in operator LineMemory. Fixed. (6436)

ROI Fix CXP Camera Operators (marathon and ironman)

Now the ROI settings work correctly with all CXP camera models. (In earlier versions, some settings for the ROI frame height lead with some specific CXP cameras to incorrect image acquisition. Fixed.) (6461)

Operator FIRoperator

Inserting nodes to operators (like BRANCH) via context menu could lead to loss of coefficients in subsequent FIRoperator modules. This has been fixed. (5450)

Mandatory Operators also in new Process 0

Designs for ironman, marathon or LightBridge: If you delete process 0 and then create a new process 0 from scratch, the mandatory operators AppletProperties and BoardStatus are inserted automatically into the new process 0 design. (In earlier versions, after removing and then re-creating process 0, the mandatory operators were missing. Fixed.) (5720)

ADD Operator

The ADD operator has been corrected so that now also two input ports with a bit width of signed 63 bit are allowed. (In earlier versions, the ADD operator did erroneously not allow two input ports with a bit width of signed 63 bit. Fixed.) (7138)

CoefficientBuffer Operator

Operator CoefficientBuffer now allows to set parameter Ylength = 1. (1763, 3924)

Parameters Library – Enhanced Control Mechanism

New error message *“DisplayName must be empty when DisplayHierarchy refers to process level”* is thrown if an operator instance of library Parameters is parameterized in a way that the referenced requested by DisplayName and DisplayHierarchy .

Simulation

- RemovePixel Operator: Simulating a design that contains an instance of the RemovePixel operator runs smoothly now. (In earlier versions, when a design was simulated that included an instance of the RemovePixel operator, occasionally an error occurred: Removing the last pixel of an image could result in an internal error that immediately stopped the simulation and returned a message about corrupted simulation data. Fixed.) (6103)
- Simulation control: Improved simulation control for situations where a single image causes generation of multiple images within an image processing pipeline. (6314)
- Pixel Data with 64 bit resolution: The calculation of simulation results for situations where pixel data with 64 bit resolution is involved works correctly now. In earlier versions, the results of such calculations were wrong. Fixed. (6878)
- Operator RamLUT now supports variable line length also when a design is simulated. The image doesn‘t need to be rectangular for simulation anymore, but can also contain empty lines and/or lines of varying length (e.g., a triangular image). (6997)

Deleting referenced Operators of library Parameters

Operator instances from library parameters can be deleted now from a design also if their parameters are still referenced by other modules. (In earlier versions, a crash of VisualApplets could happen in situations where the parameters of the deleted module were referenced by parameters of other modules (including parameter references due to property 'DisplayName'). Fixed.) (6423)

Overall Resource Estimation Improved

The overall resource estimation (displayed under *Netlist Generation* in the DRC Log after Design Rules Check 2, and in dialog *Build Hardware Applet* after starting the Build process) now also takes the RAM LUT resources into account. Therefore, a higher amount of resources is calculated than in prior VisualApplets versions where the RAM LUTs where ignored in the overall resource estimation. Now, the overall resource estimation under *Netlist Generation* is consistent with the information of the *FPGA Resource Usage* dialog. (6487)

Generating GenICam XML code for eVA Platforms

Fixed crash of VisualApplets when generating GenICam XML code (applies only to embedded VisualApplets targets) for a design which contains ColorTransform operator with dynamical coefficients. (6625)

Operator Reference (Documentation): Links to Examples of Use

Operator reference documentation: Links included in section "*Examples of Use*" now link directly to the according examples as intended. (7086)

Loading and Unloading Applets in a Loop

Loading and unloading applets (built with VisualApplets 3.0.6 or higher) in a loop doesn’t lead to memory leaks anymore. (When applets built with earlier versions were loaded and unloaded in a loop, memory leaks could be observed. Fixed.) (6925)

Available Patches

The installer(s) listed here include fixes of critical issues that became known after release of VisualApplets version 3.0.6.

Patches are distributed by Silicon Software in form of complete VisualApplets installers. The installer's file name contains a five-digit number (patch ID) that indicates which patch is included in a specific installer. For example, VAsetup3.0.6**-68118**.exe is an installer for VisualApplets 3.0.6 including patch 68121.

The installer with the highest patch ID contains all fixes available for VisualApplets release 3.0.6.

To get the latest installer, contact our Support Department: support@silicon-software.de.

Patch ID	Fixed Issue	Installer Names with Patch ID	Available since
68118	When an applet designed with VisualApplets 3.0.6 was saved as *.mcf file (via microDisplay or SDK function Fg_saveConfig), this *.mcf file provoked a crash when it was loaded via SDK function Fg_InitConfig. This has been fixed. (7333)	VAsetup3.0.6-68118.exe	11.09.2017

VisualApplets Version 3.0.4

New Features

- **New Operator ActionCommand (mE4 VQ4 only)**
  The hardware specific library for microEnable 4 VQ4-GE and microEnable IV VQ4-GPoE has a new operator *ActionCommand.* The operator allows the communication of GigE Vision action commands to and from the camera via firmware.[3]

Changes and Bug Fixes

- **CLHS Camera Discovery**
  VisualApplets CHLS designs with only one camera port: When using the final applet in a runtime environment, on camera port 2 (not connected) of the frame grabber, the system erroneously detected a camera. Fixed. Only the camera actually connected to the frame grabber running such an applet is detected now.

- **Custom Operators: Constraints Files Support (VisualApplets Expert Feature)**
  Constraints files for custom operators are fully supported now and handled correctly: Constraints files that have been defined by the user are packed into the custom library. During the build of an applet (*.hap file) that contains a custom operator, the operator constraints files are unpacked. (In the last version, packing into the library and unpacking during build didn't work properly. Fixed.)

- **Custom Operators in Protected User Library Elements (VisualApplets Expert Feature)**
  Custom operators can now be used within protected user library elements. (This is the fix of known issue no. 2 in VisualApplets 3.0.3.)
- **Operator SelectROI**
  Operator SelectROI now generates an End-of-Frame tag as soon as it has received the last line in the selected ROI. (This is the fix of known issue no. 20 in VisualApplets 3.0.3.)

[3] Applets using action commands via firmware need to be used on mE4 VQ4-GE frame grabbers that have firmware version 1.8a installed and are that are connected to a host PC that runs Silicon Software runtime software version 5.4.3 or higher.

- **Debugging Library: Improved Documentation (VisualApplets Expert Feature)**
  The documentation of the debugging library (released in VisualApplets 3.0.3 Expert) has been enhanced.

System Requirements

PC and Operating System:

- OS: Microsoft Windows 7 or Windows 8 (64bit)
- PC Memory: Minimum 4 GByte, recommended: 8 GByte or better
- Minimum available hard disk space: 500 MByte

Third-Party Software:

- For compiling the SDK examples, a C++ Compiler is necessary.
- Depending on the target platform (FPGA type) of the applets designed with VisualApplets, Xilinx Vivado® (WebPACK [free] or Design Suite) or Xilinx ISE® (WebPACK [free] or Design Suite). For details, see documentation.

Firmware Versions and Runtime Requirements

To use applets designed with VisualApplets 3.0.4, the specified minimum runtime version (or higher) must be installed on the target system.

microEnable 5 and LightBridge: In microEnable 5 and LightBridge applets, VisualApplets 3.0.4 incorporates the following firmware versions:

- LightBridge VCL: 1.4 (minimum runtime: 5.4.0)
- microEnable 5 marathon VF2: 2.0 (minimum runtime: 5.4.2)
- microEnable 5 marathon VCL: 1.4 (minimum runtime: 5.4.1)
- microEnable 5 ironman VQ8-CXP6D: 2.9 (minimum runtime: 5.4.0)
- microEnable 5 ironman VQ8-CXP6B: 2.9 (minimum runtime: 5.4.0)
- microEnable 5 ironman VD8-PoCL: 1.4 (minimum runtime: 5.4.0)

microEnable IV: Designs developed with VisualApplets 3.0.4 should be used on mE IV frame grabbers together with the following firmware versions (or higher):

- microEnable IV VD1-CL: 1.4d (hex) (minimum runtime: 5.2.1)
- microEnable IV VD4-CL: 1.34 (hex) (minimum runtime: 5.4.1)
- microEnable IV VD4-PoCL: 1.34 (hex) (minimum runtime: 5.4.1)
- microEnable IV VQ4-GE: 1.8a (hex) (minimum runtime: 5.4.1[4])
- microEnable IV VQ4-GPoE: 1.8a (hex) (minimum runtime: 5.4.1[5])

[4] When the applet contains triggers for action commands via firmware, runtime version 5.4.3 or higher is required.

[5] When the applet contains triggers for action commands via firmware, runtime version 5.4.3 or higher is required.

VisualApplets Version 3.0.3

New Features

VisualApplets Expert (Extension)

VisualApplets Release 3.0.3 for the first time includes the *VisualApplets Expert* extension[6]. As the name suggests, this extension offers a broad range of completely new expert features valuable for highly advanced users. The most important new features are:

- **Custom Operator Libraries**
  With the completely new VisualApplets *Custom Operator Functionality*, image processing modules designed in VHDL or Verilog can be used in VisualApplets as operators. The modules are incorporated as pre-synthesized IP core netlists. Each IP core builds one operator. After a GUI driven integration procedure, these operators work like built-in VisualApplets operators.

- **Debugging Library**
  The *Debugging* library offers a variety of operators with enhanced debugging capabilities. They can be used for debugging *VisualApplet Designs* in the real runtime environment as well as for debugging *custom operators* which are still under development. In the latter case, the *Debugging* library allows to analyze how new custom operators effect the designs they are instantiated in.

- **Parameters Library**
  Library *Parameters* allows to set paths to specific parameters which may be deeply embedded in hierarchical structures. This way, even parameters in protected hierarchical boxes (or in protected hierarchical boxes that are nested in other protected hierarchical boxes) can be set and reset during design time and runtime, since their parametrization interface is lifted up in the hierarchy to a hierarchical level that is not access restricted. The translation operators of the library in addition allow translated access (read and write) to

[6] To get access to the functionality of the *Expert* extension, you need an according license key.

parameters of one or several other operators. For calculation, formulas can be defined according to the GenICam API 2.0 standard.

Embedded VisualApplets (Extension)

VisualApplets Release 3.0.3 for the first time includes *eVA Designer* which is part of the *VisualApplets Embedder* extension[7]. With *eVA Designer ,* you can set up hardware platforms with image processing FPGA[8] to be programmable with VisualApplets. eVA Designer supports you in defining and testing a custom VisualApplets IP core for your FPGA and generates a dedicated plug-in for VisualApplets that introduces all details of the IP core and surrounding FPGA design to VisualApplets.

Other New Features (All Licenses)

Support of New Frame Grabber Hardware mE5 marathon VF2

- VisualApplets 3.0.3 supports programming for the new CLHS frame grabber microEnable 5 marathon VF2.

CXP Implementation

- **Flexible CXP Implementation**
  The CXP interpretation implemented in VisualApplets is more flexible now and tolerates small derivations of the CXP standard in incoming image data. This leads to improved runtime stability. The extended CXP flexibility is implemented for all CXP target platforms.

Operators

- **New Operator PixelReplicator**
  Library *Synchronization* has a new operator *PixelReplicator* that allows to replicate input pixels. The operator is available for all target hardware platforms.

[7] To get access to the functionality of the *VA Embedder* extension, you need an according license key.

[8] like cameras or other devices with imaging sensor

- **Library Logic: New Parameter in Comparison Operators**
  All comparison operators of the Logic library have a new parameter *ImplementationType* that allows the user to influence the implementation of the operator, i.e., to define which logic elements are used for implementing the operator. The comparison operators are:
  < CMP_AgeB>, <CMP_AgtB>, <CMP_AleB>, <CMP_AltB>, <CMP_Equal>, <CMP_NotEqual>, <IS_Equal>, <IS_GreaterEqual>, <IS_GreaterThan>, <IS_InRange>, <IS_LessEqual>, <IS_LessThan>, <IS_NotEqual>

- **Parameter *PcieInterfaceType* for marathon and LightBridge:**
  Operator *AppletProperties* now offers parameter *PcieInterfaceType* for all marathon and LightBridge devices. The parameter PcieInterfaceType allows you to select which PCIe interface type is supported by the applet. The selected value { Generation_1, Generation_2} defines the supported DMA bandwidth (reflected in operator DMAtoPC).

Changes and Bug Fixes

Implemented Firmware

- **Further Enhancements of PoCL Detection Mechanism:**
  - The PoCL detection mechanism implemented in VisualApplets for marathon Camera Link and LightBridge frame grabbers has been further extended. A new state of the PoCL controller has been introduced: State NonPoCL_Save is a securing time gap between sensing the PoCL compatibility of a camera and connecting GND.
  - New parameter values in parameters PoCLStatePortA/PoCLStatePortB (operator BoardStatus):
    - State NonPoCL_Save is reflected as new value of parameter PoCLStatePortA/PoCLStatePortB in operator BoardStatus.
    - State PoCL_Cable_Connection_Stability (introduced in VisualApplets 3.0.2 as 1.8 seconds for verifying the stability of the PoCL compatible signal) is now also reflected as a new value of parameter PoCLStatePortA/PoCLStatePortB in operator BoardStatus.

  Parameters PoCLStatePortA/PoCLStatePortB are dynamic-read parameters, i.e., the state of the PoCL controller can be read during runtime.

- **Enhanced PCIe Detection:**
  - All microEnable 5 marathon, ironman, and LightBridge firmware: Enhanced PCIe detection
- **Support of PCIe Generation 1:**
  - Now all microEnable 5 marathon and LightBridge firmware also supports PCIe generation 1 (in addition to PCIe generation 2). The implementation of generation 1 consumes less resources than the implementation of generation 2 and is therefore helpful in designs that do not need the high bandwidth of generation 2, but are short of resources.

Operators

- **Faster Loading of Initialization File for RamLUT Operator in Binary Mode:**
  The loading speed of the initialization file for operator RamLUT in binary mode has been enhanced.

- **RamLUT Operator: Extended Data Output**
  Operator RamLUT now supports 128 Bit Data output which allows to use the maximum RAM bandwidth.

- O-type operators can be implemented in parallel now also when they get their image data from an infinite source.

- Operators TXLink / RXLink don't cause timing errors anymore when using small parallelisms and bit widths. (was relevant to microEnable IV VD4-CL/-PoCL and microEnable IV VQ4–GE/-GPoE. Fixed.)

- Operators ROM and LUT have a new parameter "ImplementationType" that influences the implementation strategy of the operator, i.e., which logic elements are used for instantiating the operator. Available options are AUTO, BRAM, or LUTRAM. This allows to design applets using large LUTs with, e.g., 2^16 addresses on a mE5-MA-VCL (Kintex).

- SYNC operator now accesses valid memory areas also when in SyncToMax mode.

- Operators TxLink and RxLink: Enhanced documentation stating recommended settings for bit width and parallelism to ensure good routing results.

- Adding/deleting nodes on operators, hierarchical boxes, or user library elements is now possible without any side effects. All simulation modules connected to the changed operator/HBox stay connected.

- Update of user library elements is now possible without any side effects. All simulation modules connected to the updated element stay connected.

- Fix of operators CameraGrayFull und CameraRgbFull: Both operators had a bug in 80bit mode AreaScan(8tapx10bit und 10tapx8bit) that in some exceptional situations resulted in the loss of a line within a frame. (LineScan configurations where not affected.) Affected target platforms were ironman and marathon Camera Link frame grabber. Fixed.

Other Fixes

- Synthesis with Vivado®[9] tool chain also works when the VisualApplets installation path contains space characters.
- Comment boxes in designs now accept UTF-8 characters, i.e., users can add comment boxes now containing, e.g., Korean or Chinese language.
- In the Project Info panel of the VisualApplets program window, Resources: Now also the consumption of resource "Embedded ALU" is stated.
- Loading and refreshing user libraries works smoothly now.

System Requirements

PC and Operating System:

- OS: Microsoft Windows 7 or Windows 8 (64bit)
- PC Memory: Minimum 4 GByte, recommended: 8 GByte or better
- Minimum available hard disk space: 500 MByte

Third-Party Software:

- For compiling the SDK examples, a C++ Compiler is necessary.
- Depending on the target platform (FPGA type) of the applets designed with VisualApplets, Xilinx Vivado® (WebPACK [free] or Design Suite) or Xilinx ISE® (WebPACK [free] or Design Suite). For details, see documentation.

[9] Registered trade mark of Xilinx Corp.

Firmware Versions and Runtime Requirements

To use applets designed with VisualApplets 3.0.3, the specified minimum runtime version (or higher) must be installed on the target system.

microEnable 5 and LightBridge: In microEnable 5 and LightBridge applets, VisualApplets 3.0.3 incorporates the following firmware versions:

- LightBridge VCL: 1.4 (minimum runtime: 5.4.0)
- microEnable 5 marathon VF2: 1.6 (minimum runtime: 5.4.2)
- microEnable 5 marathon VCL: 1.4 (minimum runtime: 5.4.1)
- microEnable 5 ironman VQ8-CXP6D: 2.9 (minimum runtime: 5.4.0)
- microEnable 5 ironman VQ8-CXP6B: 2.9 (minimum runtime: 5.4.0)
- microEnable 5 ironman VD8-PoCL: 1.4 (minimum runtime: 5.4.0)

microEnable IV: Designs developed with VisualApplets 3.0.3 should be used on mE IV frame grabbers together with the following firmware versions (or higher):

- microEnable IV VD1-CL: 1.4d (hex) (minimum runtime: 5.2.1)
- microEnable IV VD4-CL: 1.34 (hex) (minimum runtime: 5.4.1)
- microEnable IV VD4-PoCL: 1.34 (hex) (minimum runtime: 5.4.1)
- microEnable IV VQ4-GE: 1.89 (hex) (minimum runtime: 5.4.1)
- microEnable IV VQ4-GPoE: 1.89 (hex) (minimum runtime: 5.4.1)

Known Issues

1. User-independent installation of VisualApplets in directory programs (Windows) results in access problems.
2. Custom operators cannot be used in protected user library elements.
3. When you build an applet using the Xilinx Vivado Tools, you may get a lot of critical warnings. These can be ignored.
4. Directory and file names: For naming directories and files, only fonts based on ASCII characters can be used; this means that, e.g., Asian, Cyrillic, Greek, or Arabic fonts are not supported for directory and file names.
5. Bandwidth Dialog in designs for microEnable 5 marathon and LightBridge Pre-Production frame grabbers: The values displayed for memory-based operators are not reliable. The actual data throughput of the memory operators may differ because the bandwidth analysis doesn't factor in the shared RAM concept implemented in marathon and LightBridge frame grabbers. If an operator shares the RAM with other operators, this is not detected by the bandwidth analysis and therefore is not reflected in the displayed values.
6. Only the first started instance of VisualApplets is able to save its configuration. All VisualApplets instances that have been started later have only a temporary configuration which will be discarded when the instance is closed. This concerns, e.g., build settings, library settings, system settings, general VisualApplets settings.
7. Hierarchical Boxes: If you are using hierarchical boxes, in some specific situations, the Design Rule Check may come up with the following error message: "The input XYZ of the operator ABC (hierarchical box) must be connected to an O-type operator, e.g., NOP." The reason is that some M-type operators placed within a hierarchical box cannot be connected to the input port of the hierarchical box directly. This is only true for some specific M-type operators. You can solve this problem (within the hierarchical box) by placing an NOP operator between the input port of the hierarchical box and the input port of the M-type operator.
8. SDK for CXP: Accesses to the SISO_GenICam library are not generated automatically, but have to be programmed by the user.
9. Applets for microEnable 5 platforms have to be loaded onto the frame grabber via Firmware flasher tool (microDiagnostics) in order to change the applet.

10. Bandwidth analysis does not show exact values and is only an estimation. Please use this feature very carefully and run additional tests on the target hardware.
    Bandwidth calculation in case of kernel operations does not consider kernel dimensions.
11. Operators of the color library should be used carefully: Some color conversions don't work as a user would assume:
    - HSI2RGB converts HSL -> RGB ,
    - RGB2YUV converts RGB->YCbCr,
    - XYZ2LAB uses constants according to the following definitions: http://www.easyrgb.com/math.php?MATH=M2#text2
12. Operator *FIRKernelNxM* may cause processing errors in case parameter 'EdgeHandling' is set to constant, the number of columns > 2*parallelism, the number of kernel columns is an even number, and parallelism > 1. The error can be monitored at the left border of an image, where wrong pixel data is used at the kernel positions inside the frame.
13. Operator *CoefficientBuffer* cannot be used when frame height is set to 1.
14. Simulation of kernel images: Simulation image data cannot be fed from simulation sources into kernel positions which are unequal to (0,0).
15. After a simulation error has occurred, the simulation conditions need to be reset.
16. Trigger operators may cause spikes at the trigger output line during initialization phase when loading the applet onto the frame grabber.
17. Important note concerning operating systems Microsoft Windows 8 and 7 32bit/64bit, Microsoft Windows Vista 32bit/64bit and Microsoft Windows XP 64bit:
    It is necessary and recommended to define the user folder as destination folder.
    Alternatively any other folder with full access rights can be used.
18. The DMA resource indices have to start with zero and have to be consecutively numbered. This will be checked by the DRC.
19. The operator *ImageBufferMultiRoiDyn* may cause timing errors in case of very small input images.
20. Operator SelectROI: If the defined ROI is smaller than the input frame, the last Eol/Eof is only put out after the operator has received the entire input image, i.e., SelectROI does not generate an Eof when the last line in the selected ROI has been output. There exists a workaround that will split the image into chunks of the ROI height. As each chunk is completed after the height is reached the Eof signals are not delayed.

VisualApplets Version 3.0

New Features

- **Loop Operators**
  With the new operators *TxImageLink* and *RxImageLink*, it is now possible to implement loops into designs.
- **Support of Xilinx Vivado® tools**
  For all currently available marathon and LightBridge platforms, designs can be built using Xilinx Vivado® WebPack (free) or Vivado® Design Suite in the tool chain. Xilinx ISE® WebPack (free) or ISE® Design Suite can also be used.
- **VisualApplets now 64 bit Design Tool**
  The VisualApplets design software is now a 64 bit program. This entails advanced simulation and synthesis, especially of large images and sequences.
- **Custom Library** (new library type)**:** Into the Custom Library, operators provided by a user of *VisualApplets Expert*[10] can be imported.
- **Blob and Compression Library without Runtime Licenses**
  Blob library and Compression library are now incorporated as VisualApplets 3 extensions. For using these libraries in VisualApplets 3 designs, a one-time licensing is necessary (licenses are stored on the VisualApplets Dongle). Runtime licenses for platforms using applets including operators of these libraries are no longer required.
- **Fast Fourier Transformation**
  The operator for Fast Fourier Transformation (FFT) has been refined: Now any power of two value for image width between 3 bit and the platform maximum is allowed.
- **Optimized Use of FPGA Resources in microEnable 5 Platform Designs**
  Many operators have been optimized so that FPGA resource consumption, depending on the individual design, is reduced. This may also improve synthesis time in some cases.
- **GUI Improvements**
  Various dialogs have been enhanced. The VisualApplets License Management tool is now directly available from the GUI.

[10] VisualApplets Expert is a new VisualApplets extension which will be officially released in a subsequent release. At the moment, VisualApplets Expert can be used only by users who are participants of the VisualApplets Expert evaluation programm.

System Requirements

PC and Operating System:

- OS: Microsoft Windows 7 or Windows 8 (64bit)
- PC Memory: Minimum 4 GByte, recommended: 8 GByte or better
- Minimum available hard disk space: 500 MByte

Third-Party Software:

- For compiling the SDK examples, a C++ Compiler is necessary.
- Depending on the target platform (FPGA type) of the applets designed with VisualApplets, Xilinx Vivado® (WebPACK [free] or Design Suite) or Xilinx ISE® (WebPACK [free] or Design Suite). For details, see documentation.

Firmware Versions and Runtime Requirements

To use applets designed with VisualApplets 3, the specified minimum runtime version (or higher) must be installed on the target system.

microEnable 5 and LightBridge: In microEnable 5 and LightBridge applets, VisualApplets 3 incorporates the following firmware versions:

- LightBridge VCL: 1.1 (minimum runtime: 5.4.0)
- microEnable 5 marathon VCL: 1.1 (minimum runtime: 5.4.0)
- microEnable 5 ironman VQ8-CXP6D: 2.8 (minimum runtime: 5.4.0)
- microEnable 5 ironman VQ8-CXP6B: 2.9 (minimum runtime: 5.4.0)
- microEnable 5 ironman VD8-PoCL: 1.3 (minimum runtime: 5.4.0)

microEnable IV: Designs developed with VisualApplets 3.0 should be used on mE IV frame grabbers together with the following firmware versions (or higher):

- microEnable IV VD1-CL: 1.4d (hex) (minimum runtime: 5.2.1)
- microEnable IV VD4-CL: 1.34 (hex) (minimum runtime: 5.4.1)
- microEnable IV VD4-PoCL: 1.34 (hex) (minimum runtime: 5.4.1)
- microEnable IV VQ4-GE: 1.89 (hex) (minimum runtime: 5.4.1)
- microEnable IV VQ4-GPoE: 1.89 (hex) (minimum runtime: 5.4.1)

Known Issues

1. Simulation Modules (Source and Probe) cannot be renamed.
2. Only fonts based on ASCII characters are supported; this means that, e.g., Asian, Cyrillic, Greek, or Arabic fonts are not supported.
3. Bandwidth Dialog in designs for microEnable 5 marathon and LightBridge Pre-Production frame grabbers: The values displayed for memory-based operators are not reliable. The actual data throughput of the memory operators may differ because the bandwidth analysis doesn’t factor in the shared RAM concept implemented in marathon and LightBridge frame grabbers. If an operator shares the RAM with other operators, this is not detected by the bandwidth analysis and therefore is not reflected in the displayed values.
4. Adding/deleting nodes on operators, hierarchical boxes, or user library elements: When an input or output node is inserted into or removed, all simulation modules connected to this operator get disconnected.
5. Update of user library element: When an element is updated from the user library, all simulation modules connected to this element get disconnected.
6. Only the first started instance of VisualApplets is able to save its configuration. All VisualApplets instances that have been started later have only a temporary configuration which will be discarded when the instance is closed. This concerns, e.g., build settings, library settings, system settings, general VisualApplets settings.

7. Hierarchical Boxes: If you are using hierarchical boxes, in some specific situations, the Design Rule Check may come up with the following error message: “The input XYZ of the operator ABC (hierarchical box) must be connected to an O-type operator, e.g., NOP.” The reason is that some M-type operators placed within a hierarchical box cannot be connected to the input port of the hierarchical box directly. This is only true for some specific M-type operators. You can solve this problem (within the hierarchical box) by placing an NOP operator between the input port of the hierarchical box and the input port of the M-type operator.
8. SDK for CXP: Accesses to the SISO_GenICam library are not generated automatically, but have to be programmed by the user.
9. Applets for microEnable 5 platforms have to be loaded onto the frame grabber via Firmware flasher tool (microDiagnostics) in order to change the applet.
10. Bandwidth analysis does not show exact values and is only an estimation. Please use this feature very carefully and run additional tests on the target hardware.
    Bandwidth calculation in case of kernel operations does not consider kernel dimensions.
11. Operators of the color library should be used carefully: Some color conversions don’t work as a user would assume:

    - HSI2RGB converts HSL -> RGB ,
    - RGB2YUV converts RGB->YCbCr,
    - XYZ2LAB uses constants according to the following definitions: http://www.easyrgb.com/math.php?MATH=M2#text2

12. Operator *FIRKernelNxM* may cause processing errors in case parameter 'EdgeHandling' is set to constant, the number of columns > 2*parallelism, the number of kernel columns is an even number, and parallelism > 1. The error can be monitored at the left border of an image, where wrong pixel data is used at the kernel positions inside the frame.
13. Operator *CoefficientBuffer* cannot be used when frame height is set to 1.
14. Operators *TXLink* / *RXLink* may cause timing errors when using small parallelisms and bit widths. Relevant to microEnable IV VD4-CL/-PoCL and microEnable IV VQ4–GE/-GPoE
15. Simulation of kernel images: Simulation image data cannot be fed from simulation sources into kernel positions which are unequal to (0,0).
16. After a simulation error has occurred, the simulation conditions need to be reset.

17. Trigger operators may cause spikes at the trigger output line during initialization phase when loading the applet onto the frame grabber.
18. Important note concerning operating systems Microsoft Windows 8 and 7 32bit/64bit, Microsoft Windows Vista 32bit/64bit and Microsoft Windows XP 64bit:
    It is necessary and recommended to define the user folder as destination folder.
    Alternatively any other folder with full access rights can be used.
19. The DMA resource indices have to start with zero and have to be consecutively numbered. This will be checked by the DRC.
20. The operator *ImageBufferMultiRoiDyn* may cause timing errors in case of very small input images.

SiliconSoftware
speed up your vision

Contact Details

SiliconSoftware GmbH

Steubenstrasse 46
D - 68163 Mannheim, Germany
Phone: +49(0)621.789 507 0
Fax: +49(0)621.789 507 10
Email: info@silicon.software
Web: https://silicon.software

SiliconSoftware Inc. (USA)

1 Tara Boulevard, Suite 200
Nashua, NH 03062, USA
Phone: +1 603 324 7172
Fax: +1 603 324 7101
Email: info@silicon.software
Web: https://silicon.software

SiliconSoftware America Inc.

3055 St-Martin Blvd., Suite 500
Laval, Qc., H7T0J3, Canada
Phone: +1 888.808 3670 Ext 101
Fax: +1 888.808 3670
Email: americas@silicon.software
Web: https://silicon.software

Document Details

Document Version: 1.0, Document Language: en (US), Last Change: August 2018

Disclaimer

While every precaution has been taken in the preparation of this manual, Silicon Software GmbH assumes no responsibility for errors or omissions. Silicon Software GmbH reserves the right to change the specification of the product described within this manual and the manual itself at any time without notice and without obligation of Silicon Software GmbH to notify any person of such revisions or changes.

Trademarks

Copyright Note