SILEXICA

How to Optimize an OpenCL Kernel Using Silexica's SLX FPGA

Xilinx's Vitis Software Platform

Jordon Inkeles
inkeles@silexica.com
October 2020

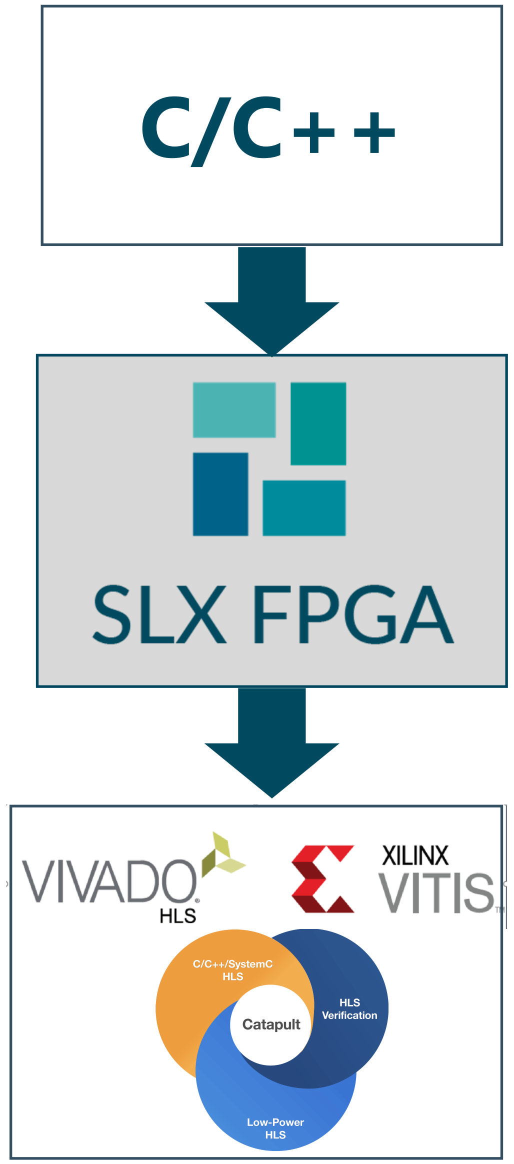
How to Optimize a Kernel in Vitis with SLX FPGA

- **Goal:** Accelerate a financial algorithm onto an FPGA from C/C++ source
- **Design:** V-model (Vasicek)– Used in the valuation of interest rate derivatives
- **Framework:** OpenCL (XRT) framework running on an Xilinx Alveo U200 Card
- **Tools:** Silexica's SLX FPGA Tool & Xilinx's Vivado/Vitis Tools

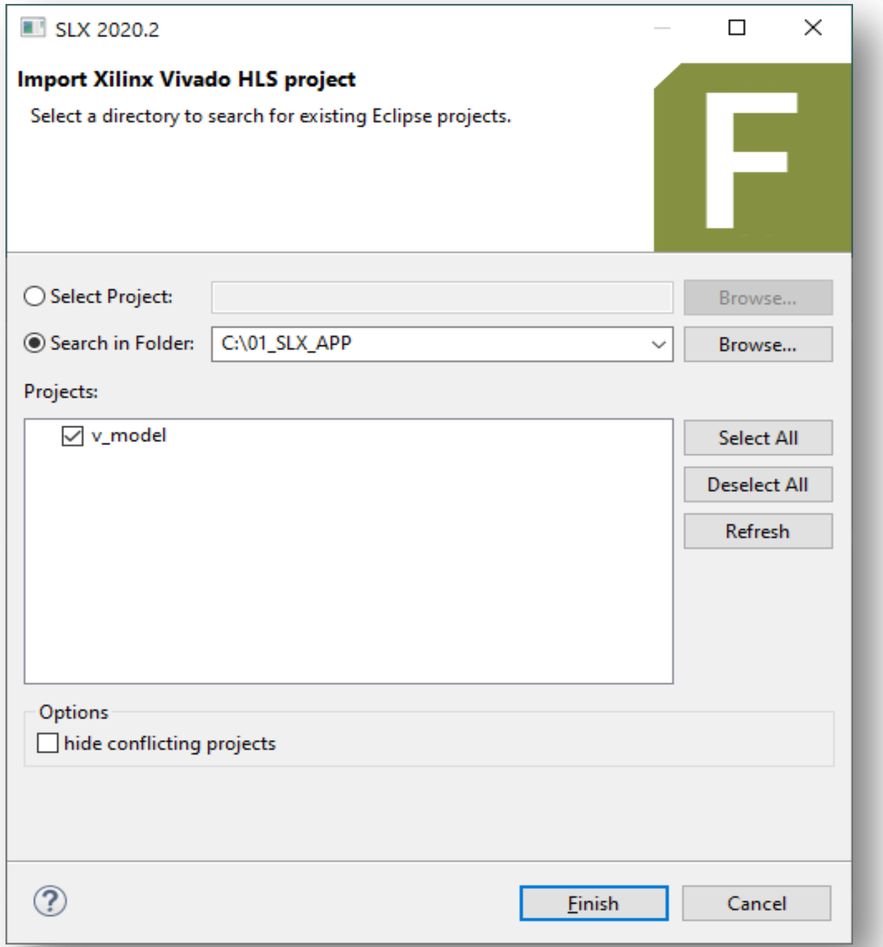
SLX FPGA

Silexica SLX FPGA

- SLX FPGA sits on top of HLS compiler
  - Prepares the C/C++ code for optimum HLS results
  - Takes the guesswork out of using HLS
- Removes the roadblocks in HLS adoption
  - Non-synthesizable C/C++ code
  - Finding parallelism
  - Poor performance and bloated area
- **HW engineers:** Get SW guidance needed
- **SW engineers:** Get parallelism/HW guidance



Import Vivado Project into SLX FPGA



SLX 2020.2

Import Xilinx Vivado HLS project

Select a directory to search for existing Eclipse projects.

( ) Select Project: [ ] Browse...

(●) Search in Folder: C:\01_SLX_APP Browse...

Projects:

- [✓] v_model

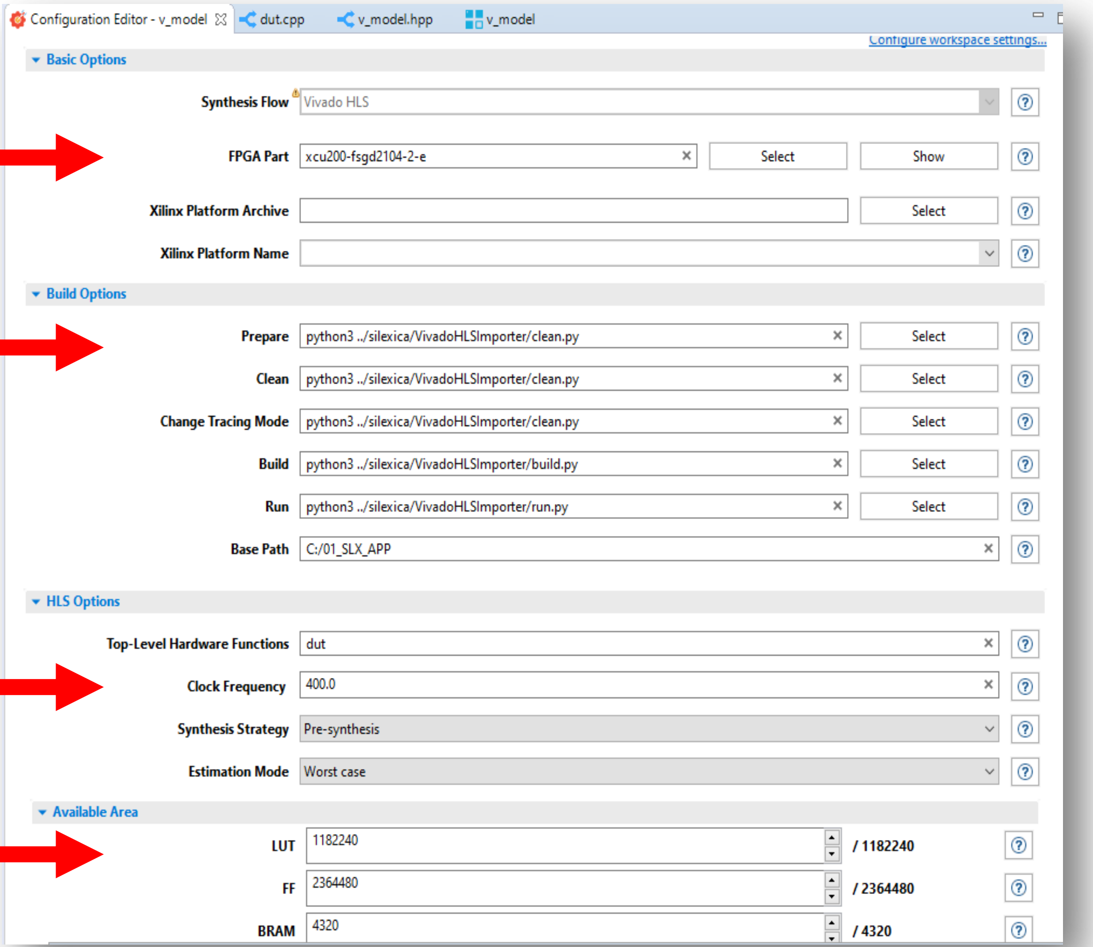
Select All | Deselect All | Refresh

Options

- [ ] hide conflicting projects

Finish | Cancel

SLX FPGA Vivado Project Importer



Configuration Editor - v_model | dut.cpp | v_model.hpp | v_model

Configure workspace settings...

Basic Options

Synthesis Flow	Vivado HLS
FPGA Part	xcu200-fsgd2104-2-e
Xilinx Platform Archive	
Xilinx Platform Name	

Build Options

Prepare	python3 ../silexica/VivadoHLSImporter/clean.py
Clean	python3 ../silexica/VivadoHLSImporter/clean.py
Change Tracing Mode	python3 ../silexica/VivadoHLSImporter/clean.py
Build	python3 ../silexica/VivadoHLSImporter/build.py
Run	python3 ../silexica/VivadoHLSImporter/run.py
Base Path	C:/01_SLX_APP

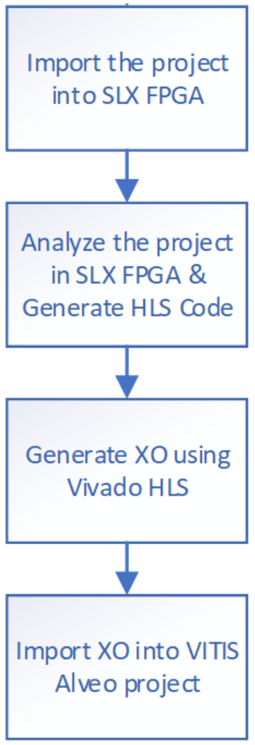
HLS Options

Top-Level Hardware Functions	dut
Clock Frequency	400.0
Synthesis Strategy	Pre-synthesis
Estimation Mode	Worst case

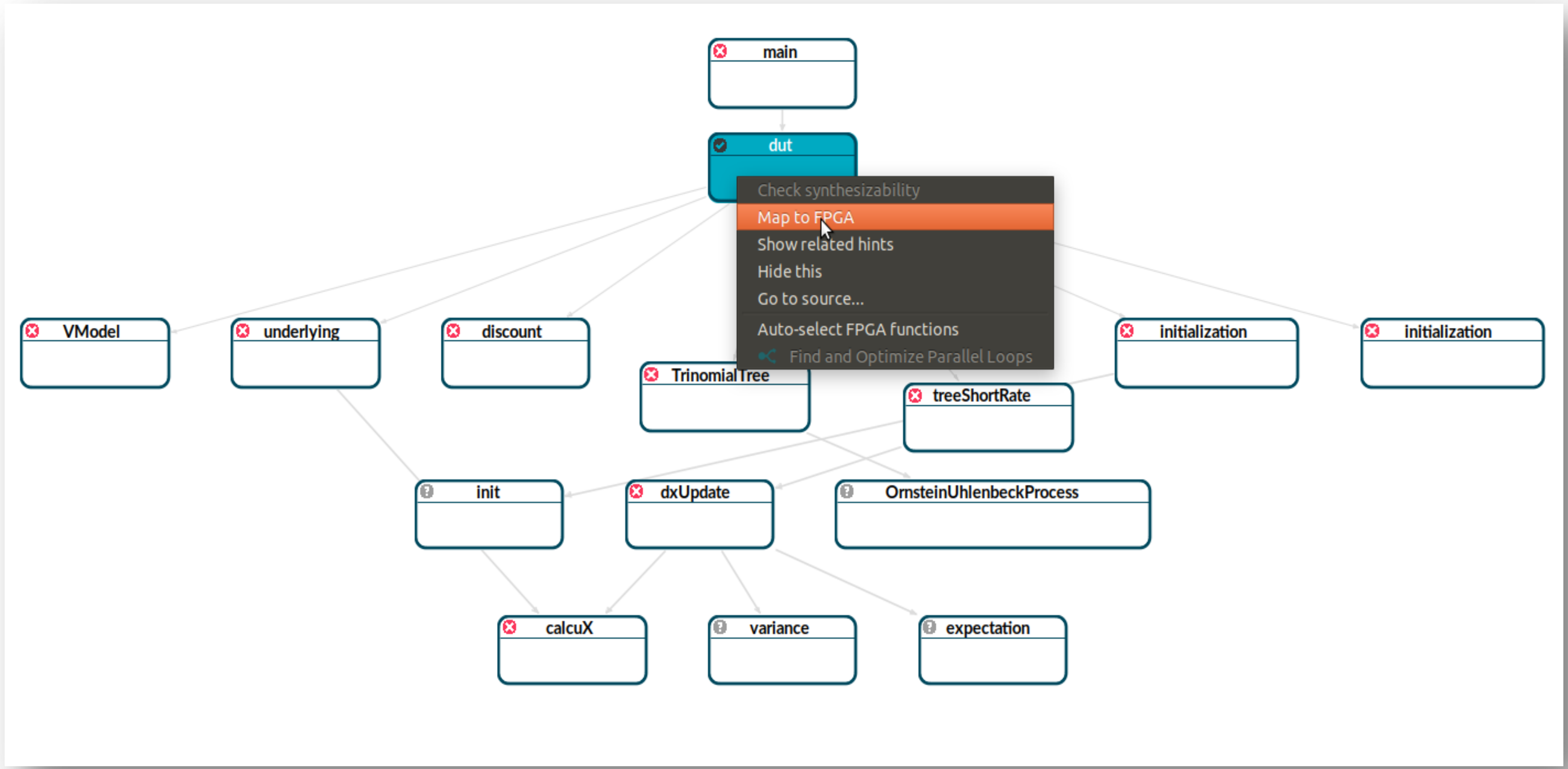
Available Area

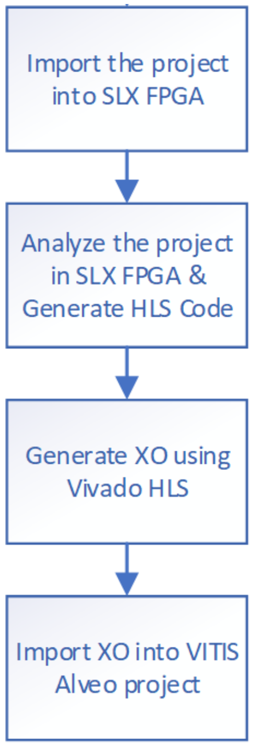
LUT	1182240	/ 1182240
FF	2364480	/ 2364480
BRAM	4320	/ 4320

SLX FPGA Configuration Editor

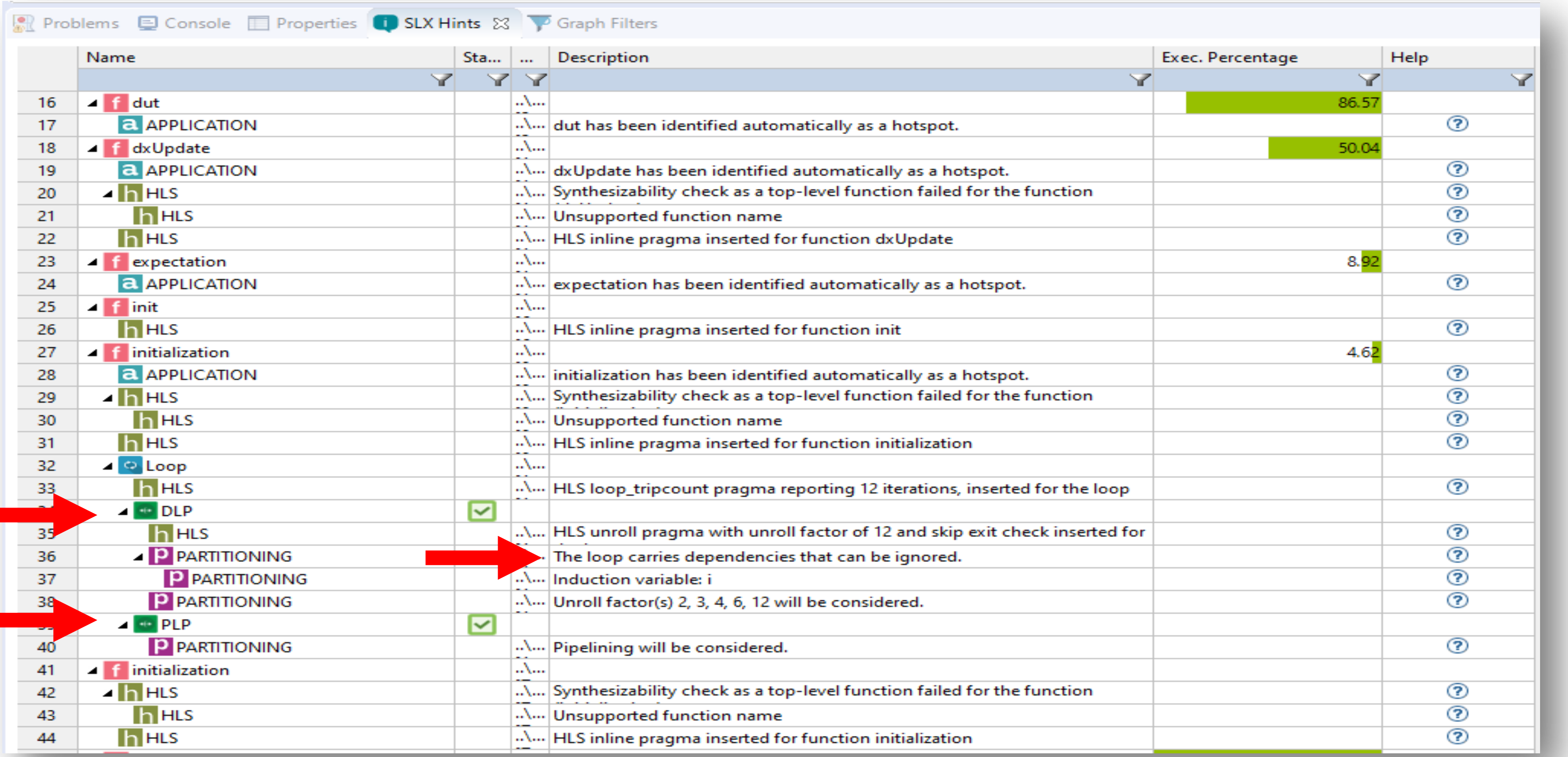

Import the project into SLX FPGA
Analyze the project in SLX FPGA & Generate HLS Code
Generate XO using Vivado HLS
Import XO into VITIS Alveo project

Analyze the Design with SLX FPGA



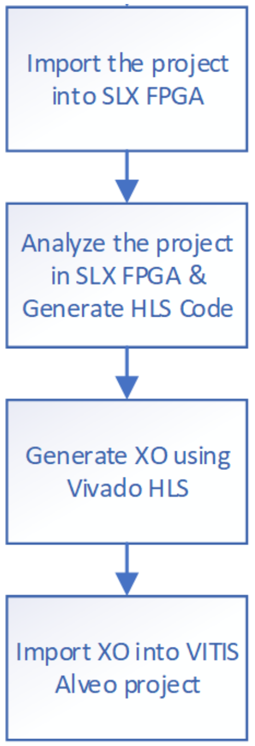

Import the project into SLX FPGA
Analyze the project in SLX FPGA & Generate HLS Code
Generate XO using Vivado HLS
Import XO into VITIS Alveo project

Analyze the Design with SLX FPGA

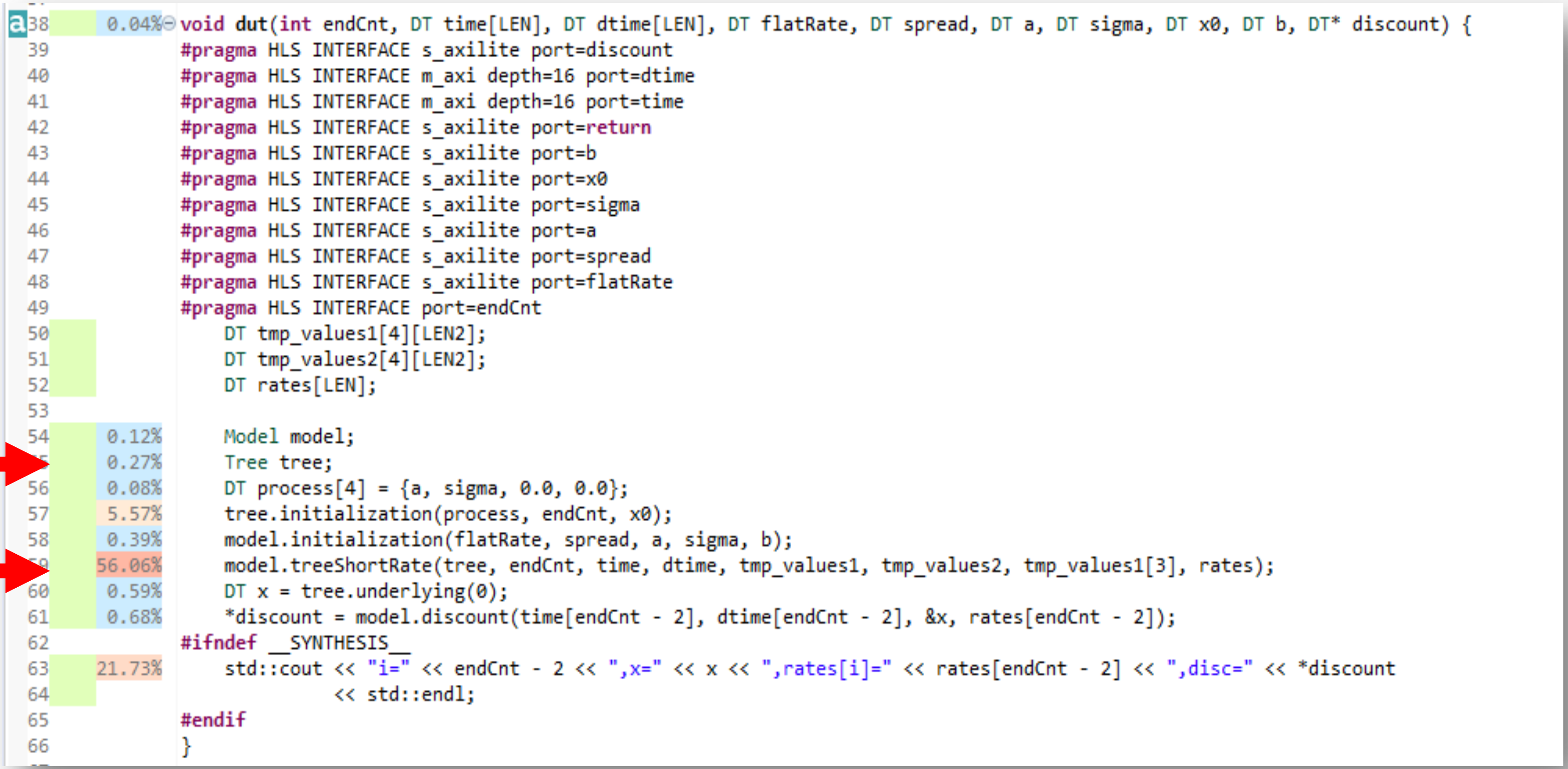


Problems | Console | Properties | SLX Hints | Graph Filters

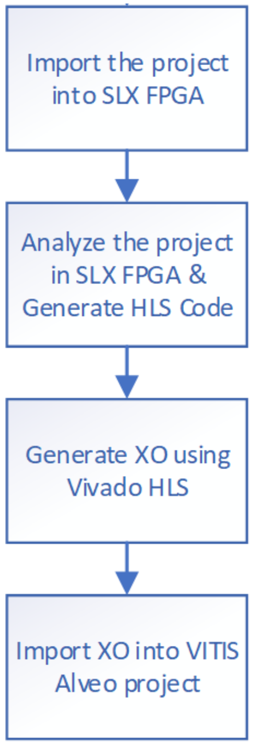
	Name	Sta...	...	Description	Exec. Percentage	Help
16	dut		..\...		86.57	
17	APPLICATION		..\...	dut has been identified automatically as a hotspot.		?
18	dxUpdate		..\...		50.04	
19	APPLICATION		..\...	dxUpdate has been identified automatically as a hotspot.		?
20	HLS		..\...	Synthesizability check as a top-level function failed for the function		?
21	HLS		..\...	Unsupported function name		?
22	HLS		..\...	HLS inline pragma inserted for function dxUpdate		?
23	expectation		..\...		8.92	
24	APPLICATION		..\...	expectation has been identified automatically as a hotspot.		?
25	init		..\...			
26	HLS		..\...	HLS inline pragma inserted for function init		?
27	initialization		..\...		4.62	
28	APPLICATION		..\...	initialization has been identified automatically as a hotspot.		?
29	HLS		..\...	Synthesizability check as a top-level function failed for the function		?
30	HLS		..\...	Unsupported function name		?
31	HLS		..\...	HLS inline pragma inserted for function initialization		?
32	Loop		..\...			
33	HLS		..\...	HLS loop_tripcount pragma reporting 12 iterations, inserted for the loop		?
[illegible]	DLP	✓				
35	HLS		..\...	HLS unroll pragma with unroll factor of 12 and skip exit check inserted for		?
36	PARTITIONING		[illegible]	The loop carries dependencies that can be ignored.		?
37	PARTITIONING		..\...	Induction variable: i		?
38	PARTITIONING		..\...	Unroll factor(s) 2, 3, 4, 6, 12 will be considered.		?
[illegible]	PLP	✓				
40	PARTITIONING		..\...	Pipelining will be considered.		?
41	initialization		..\...			
42	HLS		..\...	Synthesizability check as a top-level function failed for the function		?
43	HLS		..\...	Unsupported function name		?
44	HLS		..\...	HLS inline pragma inserted for function initialization		?


Import the project into SLX FPGA
Analyze the project in SLX FPGA & Generate HLS Code
Generate XO using Vivado HLS
Import XO into VITIS Alveo project

Analyze the Design with SLX FPGA

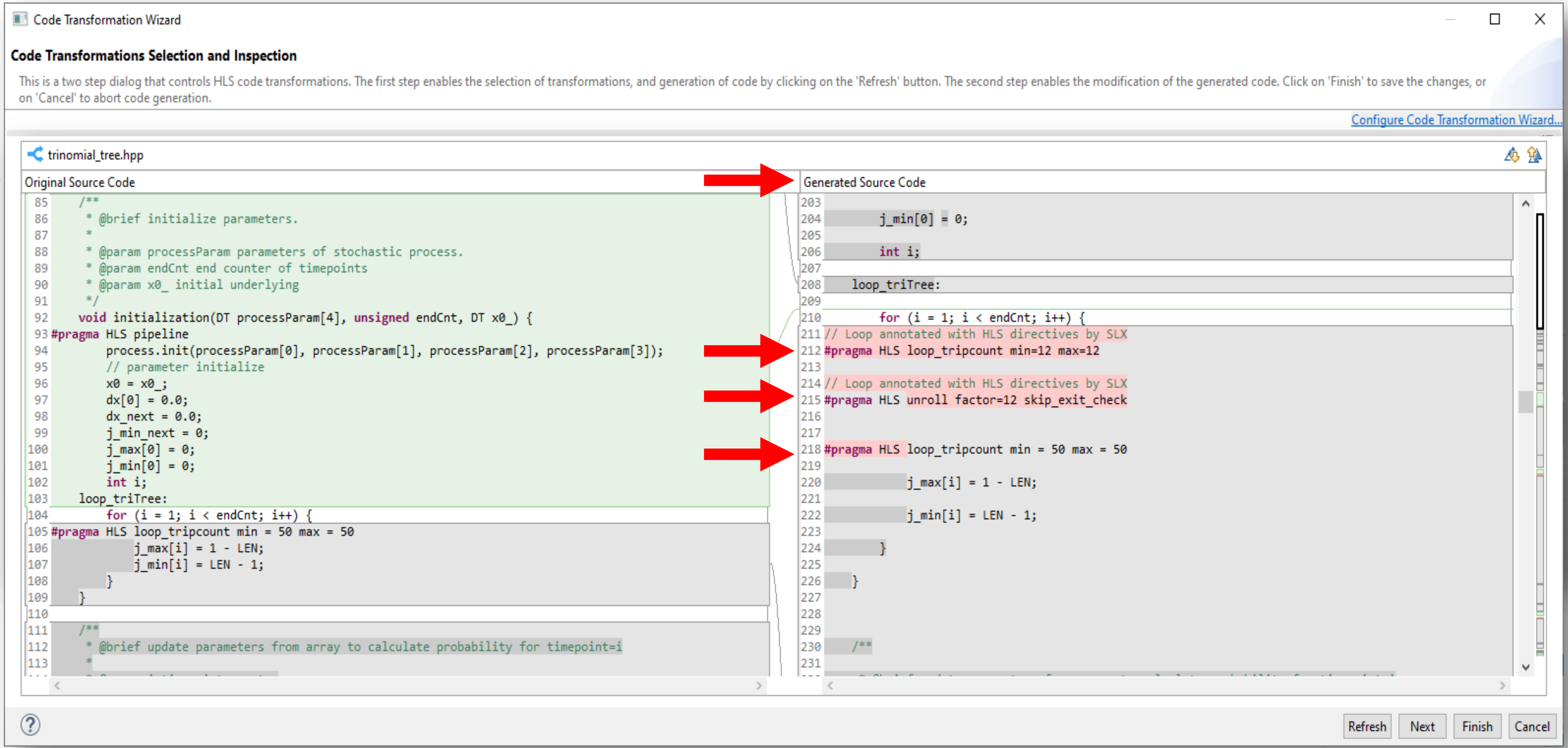


```
 0.04%  void dut(int endCnt, DT time[LEN], DT dtime[LEN], DT flatRate, DT spread, DT a, DT sigma, DT x0, DT b, DT* discount) {
        #pragma HLS INTERFACE s_axilite port=discount
        #pragma HLS INTERFACE m_axi depth=16 port=dtime
        #pragma HLS INTERFACE m_axi depth=16 port=time
        #pragma HLS INTERFACE s_axilite port=return
        #pragma HLS INTERFACE s_axilite port=b
        #pragma HLS INTERFACE s_axilite port=x0
        #pragma HLS INTERFACE s_axilite port=sigma
        #pragma HLS INTERFACE s_axilite port=a
        #pragma HLS INTERFACE s_axilite port=spread
        #pragma HLS INTERFACE s_axilite port=flatRate
        #pragma HLS INTERFACE port=endCnt
            DT tmp_values1[4][LEN2];
            DT tmp_values2[4][LEN2];
            DT rates[LEN];

 0.12%      Model model;
 0.27%      Tree tree;
 0.08%      DT process[4] = {a, sigma, 0.0, 0.0};
 5.57%      tree.initialization(process, endCnt, x0);
 0.39%      model.initialization(flatRate, spread, a, sigma, b);
56.06%      model.treeShortRate(tree, endCnt, time, dtime, tmp_values1, tmp_values2, tmp_values1[3], rates);
 0.59%      DT x = tree.underlying(0);
 0.68%      *discount = model.discount(time[endCnt - 2], dtime[endCnt - 2], &x, rates[endCnt - 2]);
        #ifndef __SYNTHESIS__
21.73%      std::cout << "i=" << endCnt - 2 << ",x=" << x << ",rates[i]=" << rates[endCnt - 2] << ",disc=" << *discount
                      << std::endl;
        #endif
        }
```



Import the project into SLX FPGA
Analyze the project in SLX FPGA & Generate HLS Code
Generate XO using Vivado HLS
Import XO into VITIS Alveo project

Automated Flow & Optimization Results


Code Transformation Wizard
Configure Code Transformation Wizard...
trinomial_tree.hpp
Refresh
Next
Finish
Cancel

Code Transformations Selection and Inspection

This is a two step dialog that controls HLS code transformations. The first step enables the selection of transformations, and generation of code by clicking on the 'Refresh' button. The second step enables the modification of the generated code. Click on 'Finish' to save the changes, or on 'Cancel' to abort code generation.

Original Source Code

```
    /**
     * @brief initialize parameters.
     *
     * @param processParam parameters of stochastic process.
     * @param endCnt end counter of timepoints
     * @param x0_ initial underlying
     */
    void initialization(DT processParam[4], unsigned endCnt, DT x0_) {
#pragma HLS pipeline
        process.init(processParam[0], processParam[1], processParam[2], processParam[3]);
        // parameter initialize
        x0 = x0_;
        dx[0] = 0.0;
        dx_next = 0.0;
        j_min_next = 0;
        j_max[0] = 0;
        j_min[0] = 0;
        int i;
    loop_triTree:
        for (i = 1; i < endCnt; i++) {
#pragma HLS loop_tripcount min = 50 max = 50
            j_max[i] = 1 - LEN;
            j_min[i] = LEN - 1;
        }
    }

    /**
     * @brief update parameters from array to calculate probability for timepoint=i
     *
```

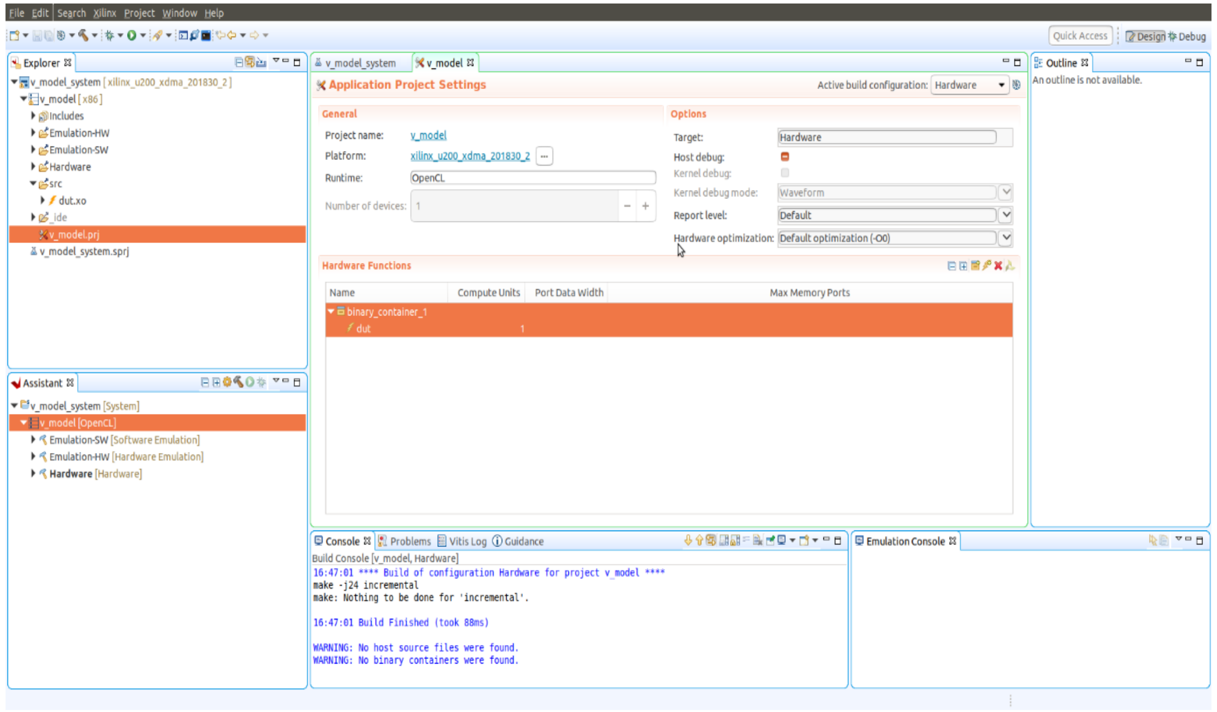
Generated Source Code

```

        j_min[0] = 0;

        int i;

    loop_triTree:

        for (i = 1; i < endCnt; i++) {
// Loop annotated with HLS directives by SLX
#pragma HLS loop_tripcount min=12 max=12

// Loop annotated with HLS directives by SLX
#pragma HLS unroll factor=12 skip_exit_check


#pragma HLS loop_tripcount min = 50 max = 50

            j_max[i] = 1 - LEN;

            j_min[i] = LEN - 1;

        }

    }



    /**
```

Automated Flow & Optimization Results

Version	Latency	LUT	FF	DSP	BRAM
Hand-Optimized (from the Vitis Library)	1823	21,939	19,891	114	12
SLX FPGA Optimized	1168	13,789	11,671	37	9
SLX Improvement	36%	38%	42%	68%	25%

Finish with Vitis Alveo project



v_model_system [xilinx_u200_xdma_201830_2]
- v_model [x86]
  - Includes
  - Emulation-HW
  - Emulation-SW
  - Hardware
  - src
    - dut.xo
  - _ide
  - v_model.prj
- v_model_system.sprj

v_model_system [System]
- v_model [OpenCL]
  - Emulation-SW [Software Emulation]
  - Emulation-HW [Hardware Emulation]
  - **Hardware** [Hardware]

Application Project Settings Active build configuration: Hardware

General

Project name: v_model
Platform: xilinx_u200_xdma_201830_2
Runtime: OpenCL
Number of devices: 1

Options

Target: Hardware
Host debug:
Kernel debug:
Kernel debug mode: Waveform
Report level: Default
Hardware optimization: Default optimization (-O0)

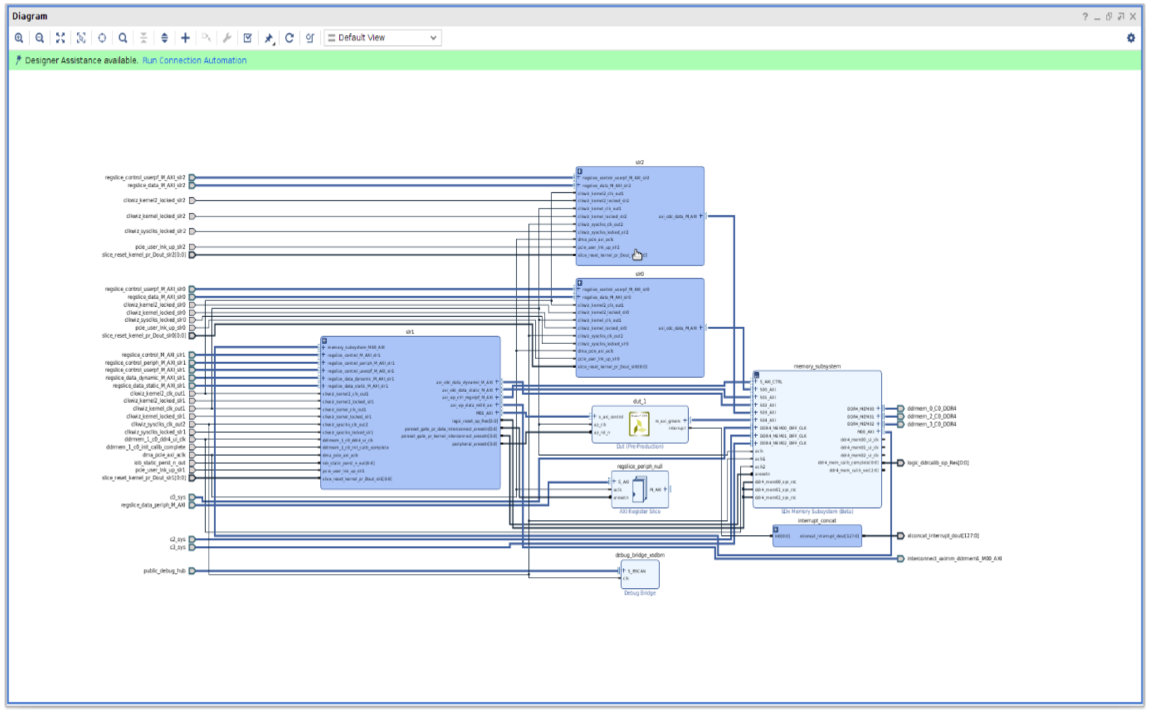
Hardware Functions

Name	Compute Units	Port Data Width	Max Memory Ports
binary_container_1			
dut	1		

An outline is not available.

Build Console [v_model, Hardware]

```
16:47:01 **** Build of configuration Hardware for project v_model ****
make -j24 incremental
make: Nothing to be done for 'incremental'.

16:47:01 Build Finished (took 88ms)

WARNING: No host source files were found.
WARNING: No binary containers were found.
```



www.silexica.com/documentation