Open Source Roadmap

The open sourcing of the NVDLA core will occur over the course of the next two calendar quarters. The intent is to deliver a useable core early, with additional configurations and features following.

The hardware design source consists of Verilog RTL. Two specific configurations are called out in this roadmap: one verified large area/performance configuration for high performace IoT applications, and one verified small area/performance configuration for cost sensitive IoT applications. The design itself will additionally be released as a configurable IP so many configurations will be possible.

The software packages will be released as source code. However, the compiler will initially be only released in binary form, with a source release coming later.

In the tables below, the term early access implies that the design source is included for the purposes of integrating into an SoC and validating that integration. However, some bugs may remain.

Components will be released at various points during the timeframes below. For a detailed listing of the released components, see the Open NVDLA Repository Updates (updates.html#updates) page.

2017-Q3

The first deliverables allow users to test with their tool flows and libraries. Both front-end simulation and back-end design will be enabled. The initial RTL will be for a large NVDLA configuration with 2048 8-bit MACs, also configurable as 1024 16-bit fixed or floating point MACs.

Detailed hardware and software documentation and an xls-based performance estimator help to determine whether NVDLA meets a user’s needs and what configuration is required.

All items listed below have been uploaded to the NVDLA GitHub repository.

Documentation	Hardware	Software
Conceptual Overview	Full-precision Config Verilog RTL, early access	Performance Estimator Tool
Hardware Architecture	Reference testbench and traces	
Software Interface	Reference synthesis scripts	
Integration Manual		
Roadmap		

2017-Q4

The second set of deliverables will add many of the tools that are needed for full system simulation. A C-model has the performance to run full networks and during this period we will also add drivers, a test application, and compilers to map any network to NVDLA.

Whereas the full-precision configuration of NVDLA was mostly focussed on performance and features, the second and small configuration will demonstrate low power and small area.

Documentation	Hardware	Software
Software Porting Guide	Full-precision config Verilog RTL released	Linux Kernel Mode Driver source
	Small config Verilog RTL, early access	QEMU based C model suitable for software development
		Linux User Mode Driver Runtime source
		Binary compiler
		Application supporting Caffe and TensorFlow models; most CNN related layers
		Software sanity tests

2018-H1

The third set of deliverables will see scalable solutions: the user can set parameters to create the exact configuration that best match his or her needs. A test bench, test suite, and FPGA support are added to allow for rigorous testing needed for high-confidence tape outs.

Documentation	Hardware	Software
Software architecture document	Small config Verilog RTL released	Refer to GitHub (https://github.com/nvdla/sw/blob/master/Roadmap.md) for software roadmap.
	RTL support for fine grained configuration control	
	UVM testbench validation of custom configurations	