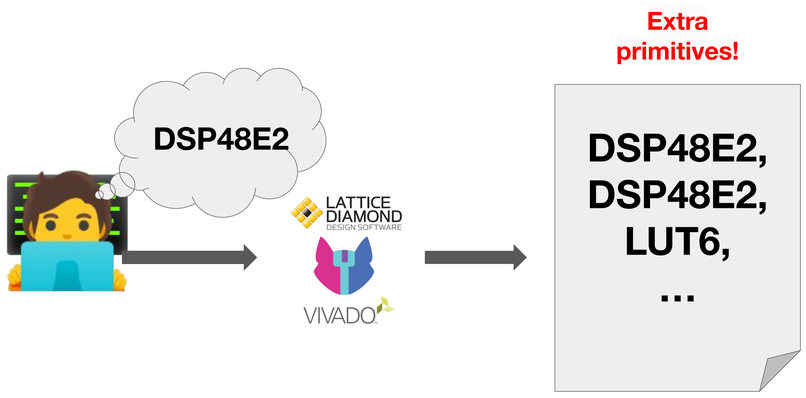
Cookie: Surveying FPGA Technology Mapping Completeness

Andrew Cheung

Hardware Compilation

FPGA hardware compilers take a high-level design and output a low-level implementation as a series of hardware primitives. FPGA primitives are quite complex; **they can efficiently implement many micro-designs**!

However, to the misfortune of the end user, hardware compilers are **incomplete**; they do not fully leverage the power of hardware. This manifests in the compiler outputting nonoptimal designs with unnecessary hardware primitives.

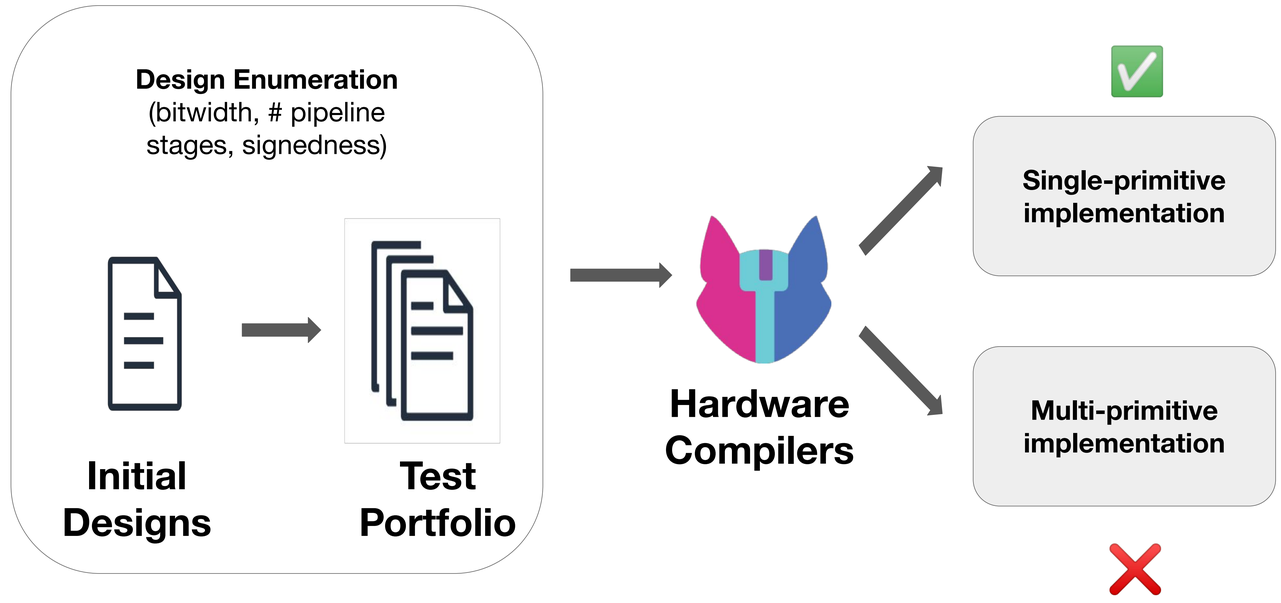


The Problem

How complete are hardware compilers? There is no existing methodology to measure this. As a result, hardware designers need to scour support forums and guess at whether the compiler will fully take advantage of hardware and output an efficient design.

Our Solution

Cookie takes an initial design and tweaks it across several dimensions, which can then be used as input to a suite of FPGA hardware compilers.



Preliminary Results

Workload	Signed?	# Stages	Yosys	SOTA	Lakeroad	Verif?	Valid?
`((d + a) * b) | c`	X	1	1 DSP, 20 LUT	1 DSP, 10 LUT	1 DSP	✓	✓
`((d - a) * b) | c`	✓	2	1 DSP, 20 LUT	1 DSP, 10 LUT	1 DSP	✓	✓
`((d - a) * b) ^ c`	✓	3	1 DSP, 22 LUT	2 DSP, 11 LUT	1 DSP	✓	✓
`((d + a) * b) & c`	✓	3	1 DSP, 22 LUT	2 DSP, 11 LUT	1 DSP	✓	✓
`((d + a) * b) ^ c`	X	2	1 DSP, 18 LUT	1 DSP, 9 LUT	1 DSP	✓	✓

PLSE UW PAUL G. ALLEN SCHOOL OF COMPUTER SCIENCE & ENGINEERING