Instruction-Level Abstraction Synthesis Template Definition

Hongce Zhang   Pramod Subramanyan   Bo-Yuan Huang   Aarti Gupta[†]   Sharad Malik
Departments of Computer Science[†] and Electrical Engineering, Princeton University

1 Template for Instructions

Template provides human insights for abstraction.

A template should specify module input W and the architecture states S including both registers and memories. These states should be visible from the port. The visibility here can be direct, meaning its value is accessible from the outside, or indirect, where some directly visible states are updated according to its value.

Fetch function F inside the template specifies what an abstract instruction makes of. It is a function that maps states and inputs to a bitvector (or the instruction). The fetch valid function V indicates the condition when there is truly instruction for the module. It is designed to help model accelerators which do not always have an instruction to execute.

Each instruction is associated with a decode function. Given the bitvector generated by fetching, it distinguishes among different instructions.

Unlike ILA, the template does not have a set of state update functions, but a unified one which maps the current vector of states and inputs to the vector of future values.

The template is defined as,

$$
\begin{aligned}
\mathrm{T} \;=\;& \langle S, W, F, V, D, P, T \rangle, \text{ where} \\
S \;\;& \text{ is the vector of state variables,} \\
W \;\;& \text{ is the vector of input variables,} \\
V \;\;& : (S \times W) \mapsto \mathbb{B} \text{ is the valid function,} \\
F \;\;& : (S \times W) \mapsto bvec_w \text{ is the fetch function,} \\
D \;=\;& \{\delta_i : bvec_w \mapsto \mathbb{B}\} \text{ is the set of decode functions,} \\
P \;=\;& \{p_j\} \text{ is the set of parameters,} \\
& \text{whose values depends solely on the assignment of } D, \text{ and} \\
N \;=\;& (S \times W \times P \times U) \mapsto S \text{ is the template of next state functions.}
\end{aligned}
$$

The parameters $\{p_j\}$ here are the "holes" we allow in the our template. They appear in the form of unknown constants within a range, undetermined choices, slicing certain bits from a bitvector and so on. The value of these parameters should be a function of an assignment to the set D of decode functions. In other word, the unknown constant, the choice, etc. should be completed fixed for a given kind of instructions.

2 Template for Child-Instructions