MONASH University

Department of Electrical and Computer Systems Engineering

ECE4095 Final Year Project, Semester 2, 2014

Reuben D'Netto

H2V – a Haskell to Verilog Compiler

Supervisor: Dr. David Boland

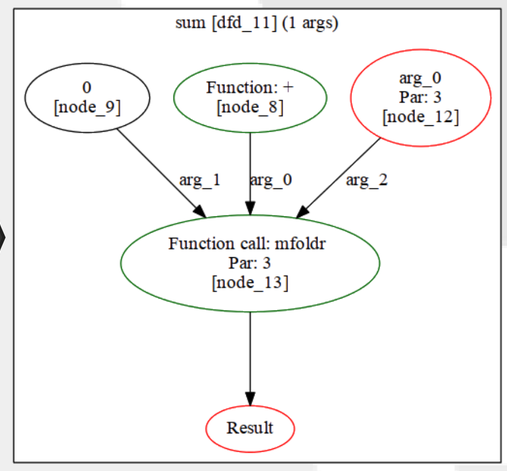
Verilog is often used to implement hardware accelerators, which are used to perform expensive computations faster than a general purpose CPU would allow.

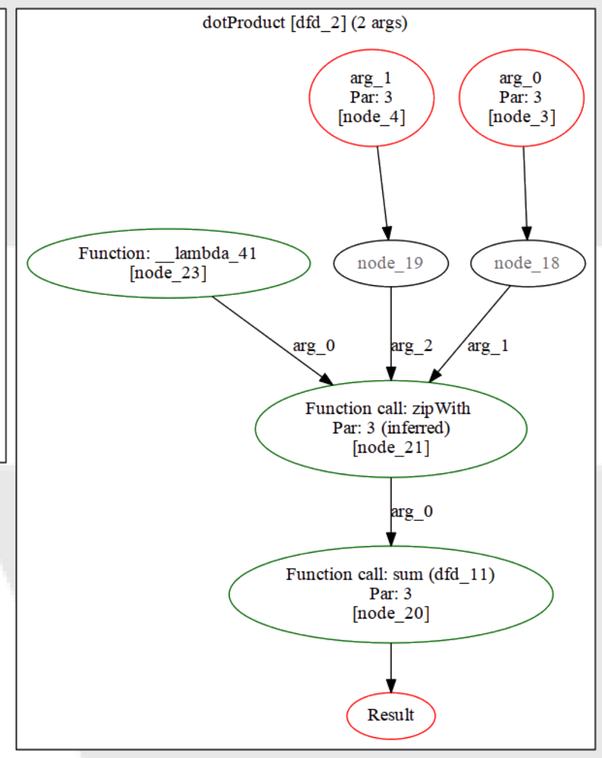
H2V generates Verilog modules from concise functional descriptions of logic, making it trivial to leverage data-level parallelism.

Logic can be tested with desktop Haskell compilers, reducing development time.

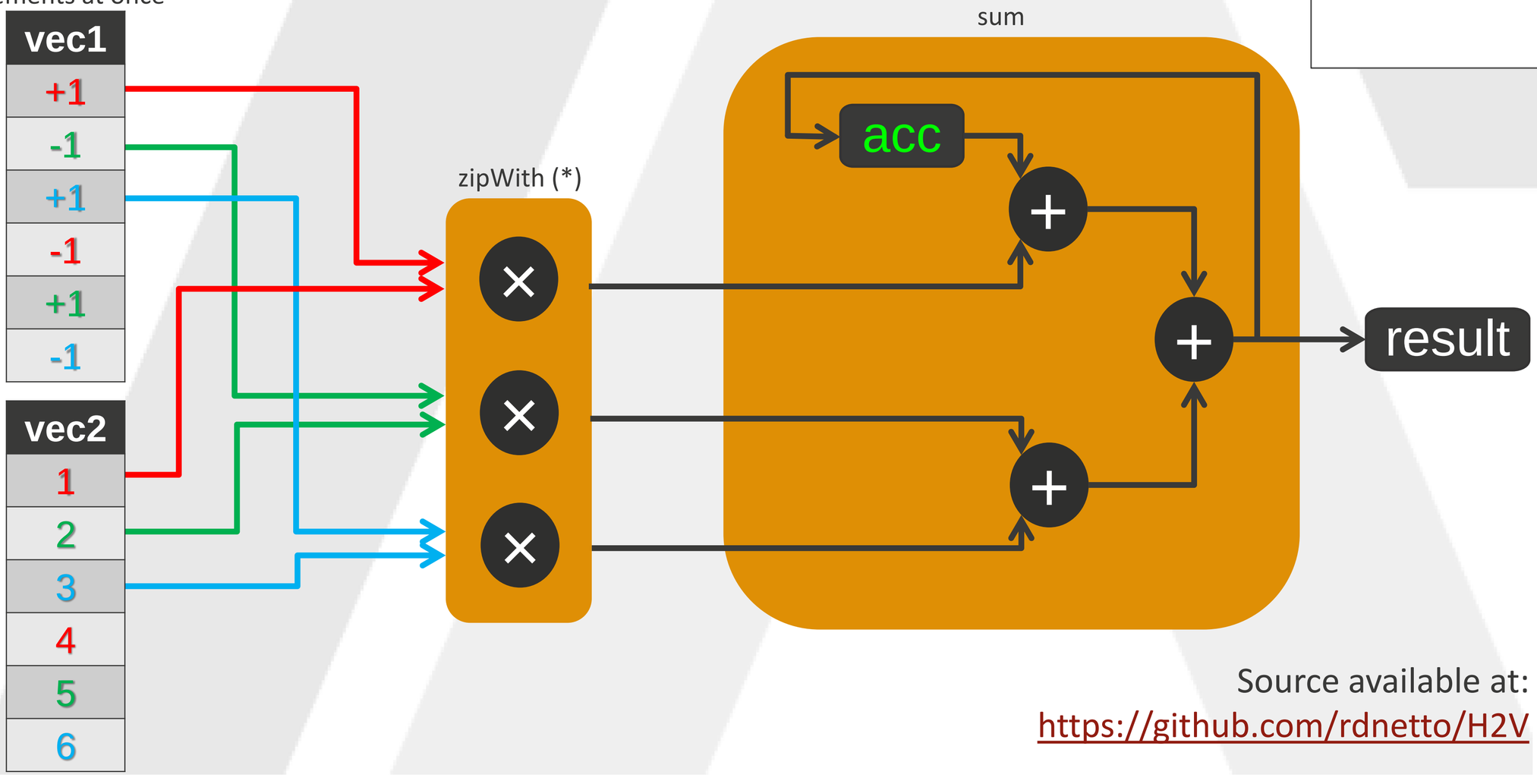
- Trivial composition of modules
- Compatible with existing Haskell compilers
- Easily tuned N-degree parallelization

```
sum :: [Int] -> Int
sum = mfoldr (+) 0 ||| 3

dotProduct :: [Int] -> [Int] -> Int
dotProduct u v = sum $ zipWith (*) u v ||| 3

demo _ = dotProduct vec1 vec2 ||| 3 where
     vec1 = [+1, -1, +1, -1, +1, -1]
     vec2 = [1 .. 6]
```





Lists – processing 3 elements at once



Source available at:
https://github.com/rdnetto/H2V