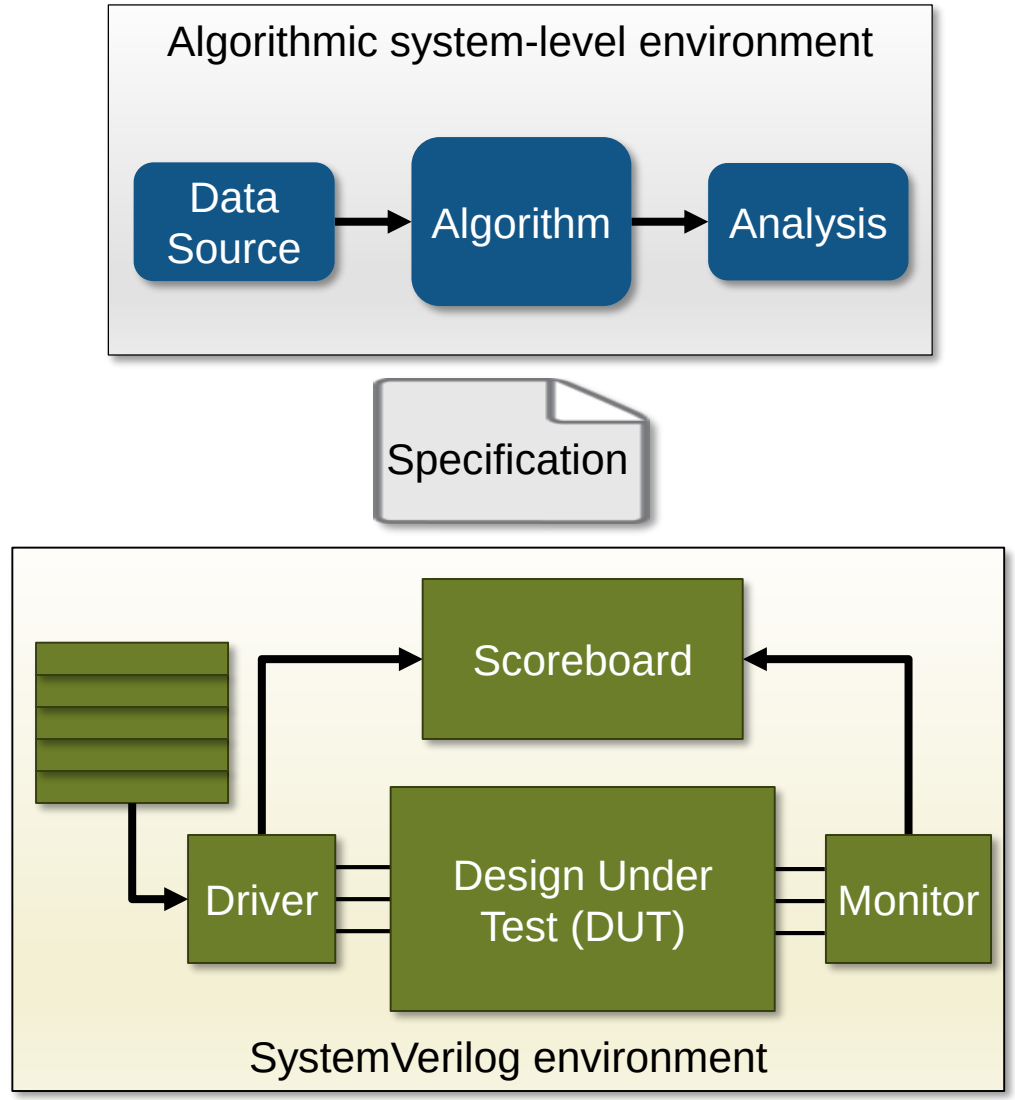
HDL Verifier DPI Component Generation from MATLAB

Reuse MATLAB functions in Universal Verification Methodology simulation environments

Verification Environment Creation

- System and algorithm verified in MATLAB or Simulink
  - Algorithmic models
  - Full system environment
  - Realistic stimulus
- Specification written, passed to verification
- Verification interprets spec to recreate:
  - Checker model
  - Models external to DUT
  - Stimulus

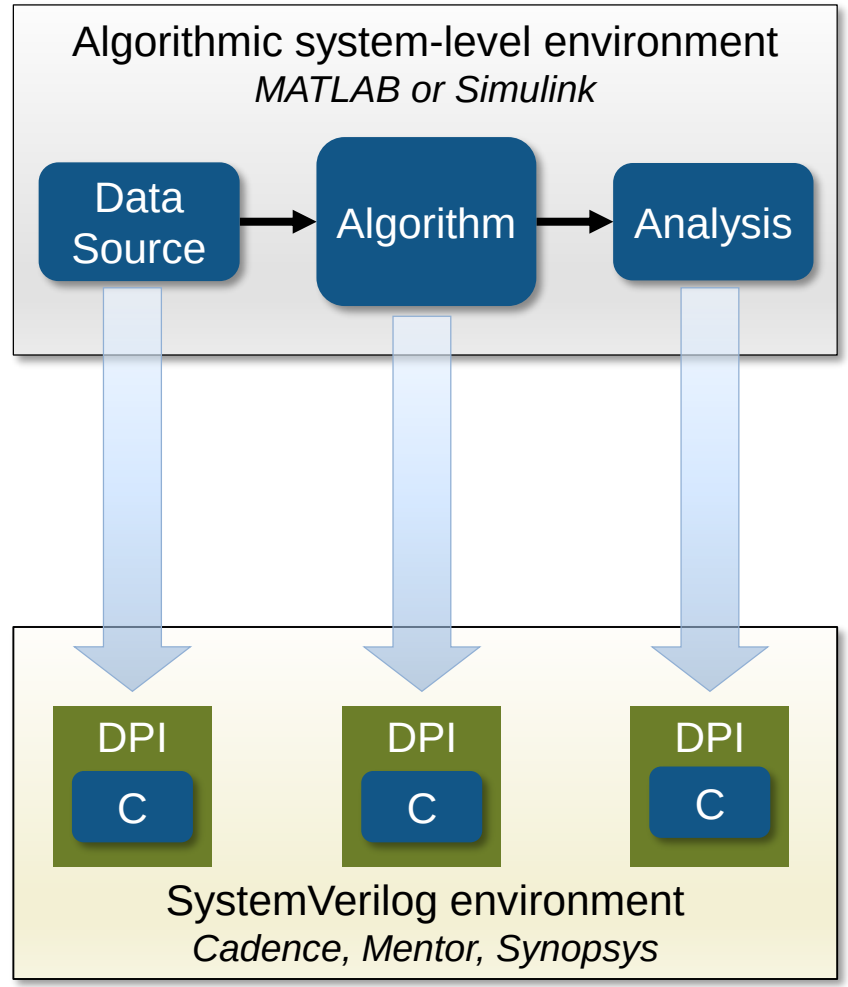
Why recreate when we can reuse?



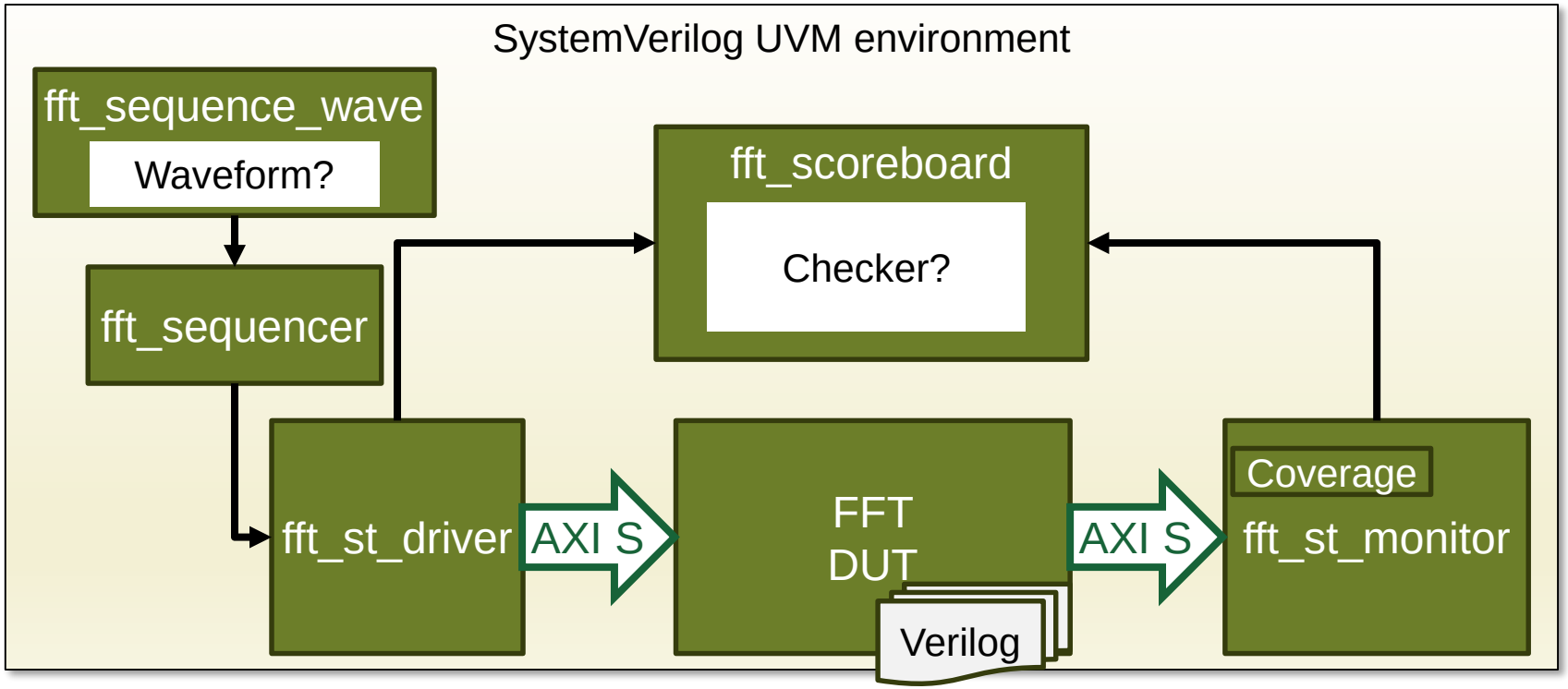
HDL Verifier

Automatically generate SystemVerilog DPI components

- Reuse MATLAB/Simulink models in verification
  - Available immediately
  - Already verified
  - Easy to update
- Everything is automated
  - Generates code and SystemVerilog interface
  - Generates and runs makefile to build shared library
- Anywhere C code can be generated
  - Digital and analog
  - Broad block and language support



Design Overview



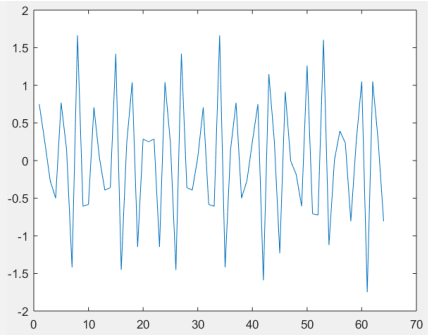
Checker:

- Calculate floating point FFT
- Calculate difference vs. DUT

$$normalized\ rms\ error\ = \frac{rms(error)}{rms(result)}$$

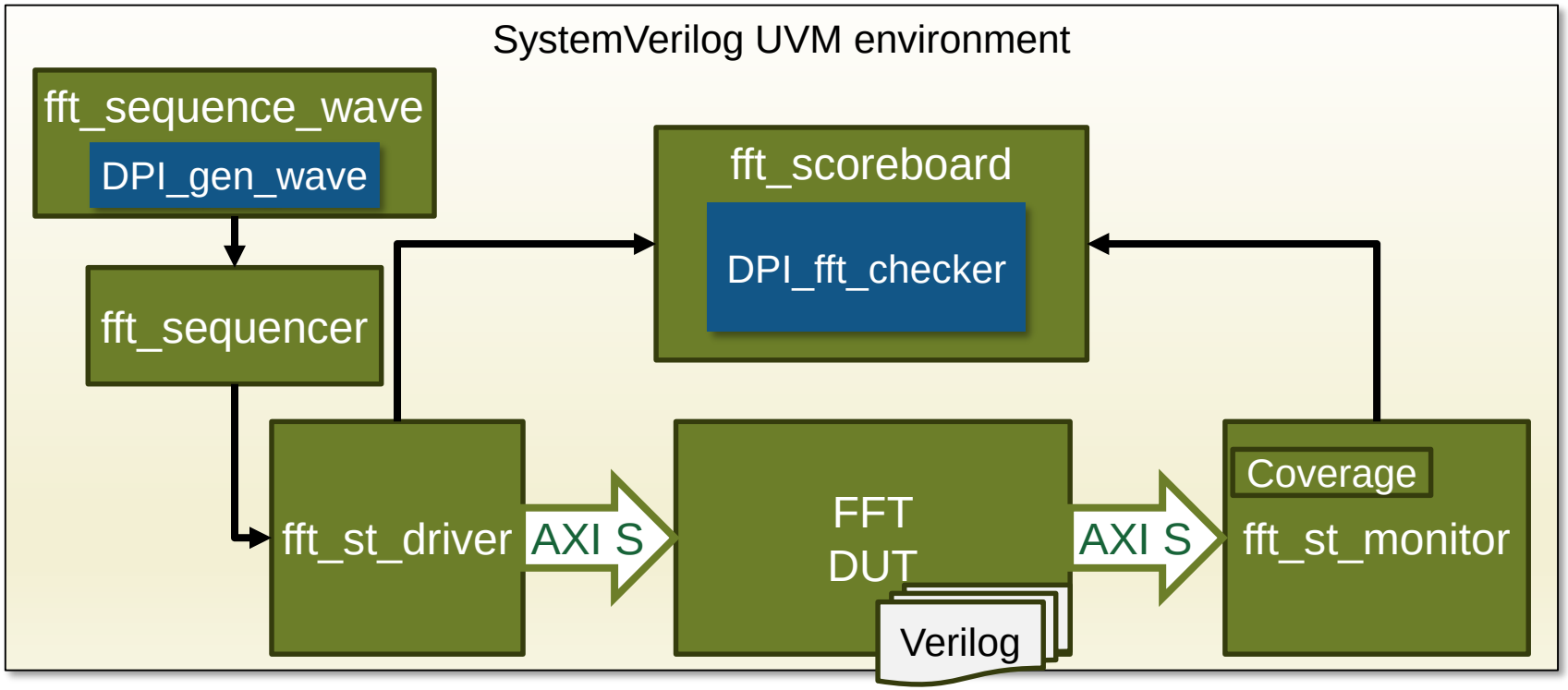
- Compare vs. theoretical upper bound

Waveform:



What would it take to write these in SystemVerilog?

Results



Checker:

- 5 lines of MATLAB
- Generated with a single command
- Easily adjusted

Waveform:

- 10-line MATLAB function
- Generated with a single command
- Easily adjusted or replicated

How much work does this save?

HDL Verifier SystemVerilog DPI Component Generation

Reuse MATLAB and Simulink models for verification

- ✓ Models available earlier
- ✓ Accurately capture algorithm behavior
- ✓ Easy to update
- ✓ Applicable to broad class of models