



A new language for hardware design

Lennart Van Hirtum

I am

- Lennart Van Hirtum
- Masters in CS at KU Leuven in Belgium
- PhD program at Paderborn
Center for Parallel Computing

My fields

- Hardware design
- High-performance Computing
- Discrete mathematics & Graph Theory
- Programming Language Design



SUS Hardware Design Language

- For directly designing hardware (No code synthesis)
- Make timing easier to reason about
- Abstractions for Pipelining
& Clock Domains
- Good developer feedback -> LSP
- Written in Rust

- LATTE'24 short talk in San Diego



SUS Examples

```
module fizz_buzz : int v -> int fb {  
  gen int FIZZ = 15  
  gen int BUZZ = 11  
  gen int FIZZ_BUZZ = 1511  
  
  bool fizz = v % 3 == 0  
  bool buzz = v % 5 == 0  
  
  if fizz & buzz {  
    fb = FIZZ_BUZZ  
  } else if fizz {  
    fb = FIZZ  
  } else if buzz {  
    fb = BUZZ  
  } else {  
    fb = v  
  }  
}
```

```
module pow17 : int i -> int o {  
  int i2 = i * i  
  reg int i4 = i2 * i2  
  int i8 = i4 * i4  
  reg int i16 = i8 * i8  
  o = i16 * i  
}
```



github.com/pc2/sus-compiler