RISC-V MYTH PROGRAM
TL-Verilog

Transaction-Level Verilog (TL-Verilog) is an emerging extension to SystemVerilog that supports transaction-level design methodology. In transaction-level design, a transaction is an entity that moves through a microarchitecture. It is operated upon and steered through the machinery by flow components such as pipelines, arbiters, and queues. A transaction might be a machine instruction, a flit of a packet, or a memory read/write. The flow of a transaction can be established independently from the logic that operates on the transaction.

Features of TL-Verilog :

1. Pipeline-Centric Design:

- TL-Verilog introduces implicit pipelines using `|pipe`, making sequential logic more structured and readable.
- It eliminates the need for explicit clocking and `always` blocks.

2. Concise and Readable Syntax:

- Reduces boilerplate code compared to traditional Verilog/SystemVerilog.
- Uses **hierarchical scoping (`|` notation)** for better organization.

3. Automatic Signal Propagation:

- Variables automatically flow across pipeline stages (`@0`, `@1`, etc.).
- No need for manually coding registers or delays.

4. Modular and Reusable Design:

- Encourages reuse of pipeline structures, reducing redundancy.
- Scoped signals prevent naming conflicts and improve clarity.

5. Enhanced Debugging and Simulation:

- Provides built-in visualization tools like Makerchip for interactive debugging.
- Debugging is simpler and more intuitive due to structured pipeline stages.

6. Clock and Reset Abstraction:

- No need to explicitly define clocks and resets; TL-Verilog handles them automatically.
- Reduces errors related to clock-domain crossings.

7. Integration with Existing Tools:

- TL-Verilog can be translated to standard Verilog, making it compatible with traditional FPGA and ASIC toolchains.
- Works with open-source tools like Yosys, Verilator, and Makerchip.

These features make TL-Verilog a modern, efficient alternative for hardware design, particularly in complex pipeline-based architectures.

Makerchip IDE: A Cloud-Based IDE for TL-Verilog

Makerchip is a **cloud-based integrated development environment (IDE)** designed specifically for **TL-Verilog**, enabling efficient digital hardware design, simulation, and debugging. It simplifies **RISC-V, FPGA, and ASIC development** by providing an interactive and user-friendly environment.

Key Features of Makerchip IDE

◈ **Web-Based and Accessible:**

- No installation required; accessible via any web browser.
- Supports real-time collaboration and sharing.

◈ **Support for TL-Verilog:**

- Built-in support for TL-Verilog syntax and hierarchical design.
- Reduces complexity by automatically handling pipelining and clocking.

◈ **Integrated Simulation and Debugging:**

- Provides real-time waveform visualization for debugging.
- Supports interactive simulation to verify designs efficiently.

◈ **Seamless Verilog Integration:**

- Converts TL-Verilog to standard Verilog for FPGA/ASIC synthesis.
- Works with open-source tools like Yosys, Verilator, and OpenROAD.

◈ **M4 Macro Preprocessor Support:**

- Enables parameterized design and macro-based code generation.
- Helps in modularizing large designs efficiently.

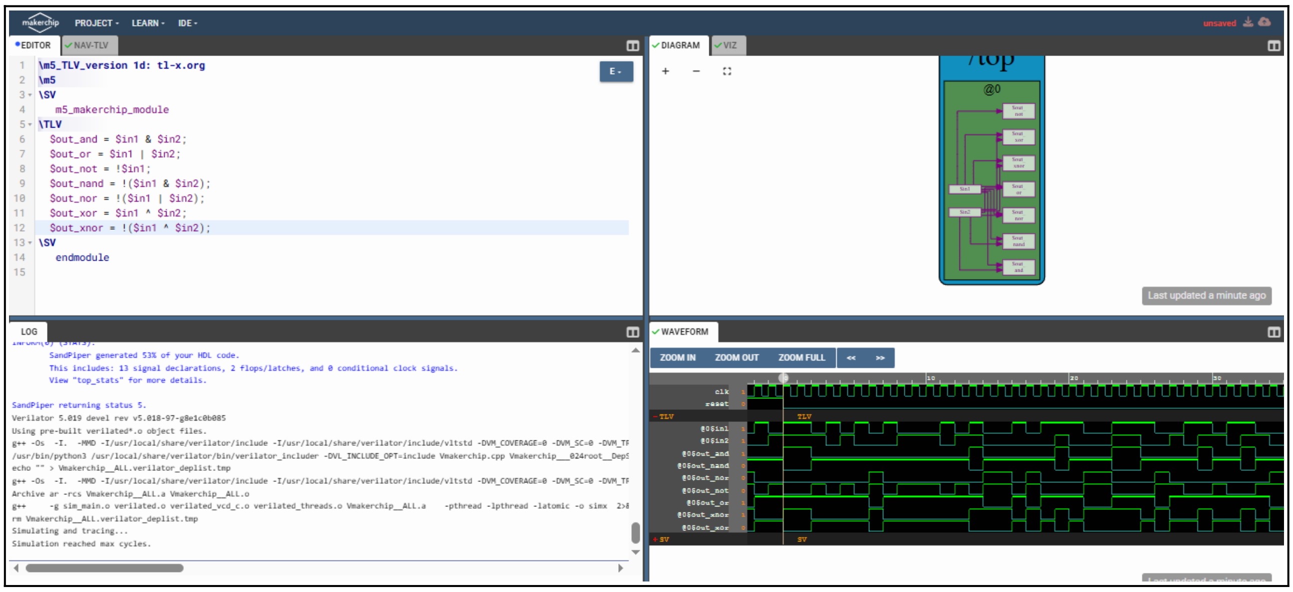
Why Use Makerchip IDE?

- Reduces development time with automated pipelines.
- Enhances readability compared to traditional Verilog.
- Ideal for RISC-V and complex digital designs.

Combinational logic in TL-Verilog :

1. Logic Gates :

```
\m5_TLV_version 1d: tl-x.org
\m5
\SV
   m5_makerchip_module
\TLV
  $out_and = $in1 & $in2;
  $out_or = $in1 | $in2;
  $out_not = !$in1;
  $out_nand = !($in1 & $in2);
  $out_nor = !($in1 | $in2);
  $out_xor = $in1 ^ $in2;
  $out_xnor = !($in1 ^ $in2);
\SV
   endmodule
```



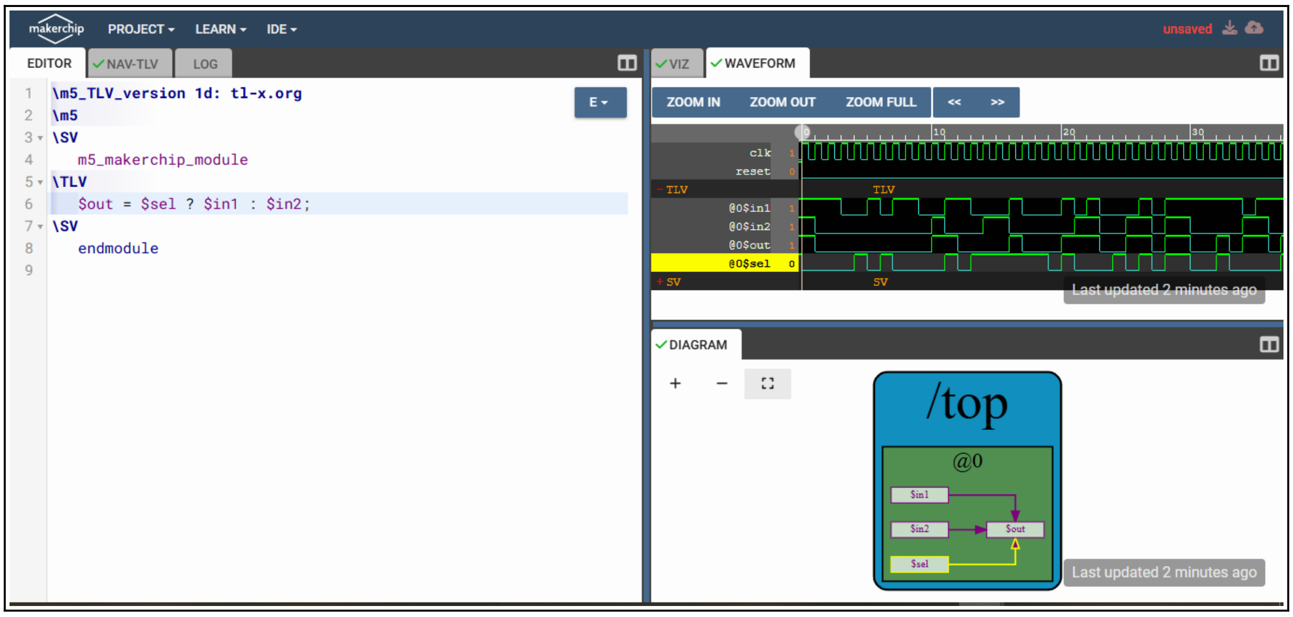
```
\m5_TLV_version 1d: tl-x.org
\m5
\SV
   m5_makerchip_module
\TLV
  $out_and = $in1 & $in2;
  $out_or = $in1 | $in2;
  $out_not = !$in1;
  $out_nand = !($in1 & $in2);
  $out_nor = !($in1 | $in2);
  $out_xor = $in1 ^ $in2;
  $out_xnor = !($in1 ^ $in2);
\SV
   endmodule
```

```
        SandPiper generated 53% of your HDL code.
        This includes: 13 signal declarations, 2 flops/latches, and 0 conditional clock signals.
        View "top_stats" for more details.

SandPiper returning status 5.
Verilator 5.019 devel rev v5.018-97-g8e1c0b085
Using pre-built verilated*.o object files.
g++ -Os  -I.  -MMD -I/usr/local/share/verilator/include -I/usr/local/share/verilator/include/vltstd -DVM_COVERAGE=0 -DVM_SC=0 -DVM_TR
/usr/bin/python3 /usr/local/share/verilator/bin/verilator_includer -DVL_INCLUDE_OPT=include Vmakerchip.cpp Vmakerchip___024root__DepS
echo "" > Vmakerchip__ALL.verilator_deplist.tmp
g++ -Os  -I.  -MMD -I/usr/local/share/verilator/include -I/usr/local/share/verilator/include/vltstd -DVM_COVERAGE=0 -DVM_SC=0 -DVM_TR
Archive ar -rcs Vmakerchip__ALL.a Vmakerchip__ALL.o
g++     -g sim_main.o verilated.o verilated_vcd_c.o verilated_threads.o Vmakerchip__ALL.a    -pthread -lpthread -latomic -o simx  2>&
rm Vmakerchip__ALL.verilator_deplist.tmp
Simulating and tracing...
Simulation reached max cycles.
```

Makerchip : https://makerchip.com/sandbox/0lYfohqE9/03lhpRr

2. 2X1 MUX :

```
\m5_TLV_version 1d: tl-x.org
\m5
\SV
   m5_makerchip_module
\TLV
   $out = $sel ? $in1 : $in2;
\SV
   endmodule
```

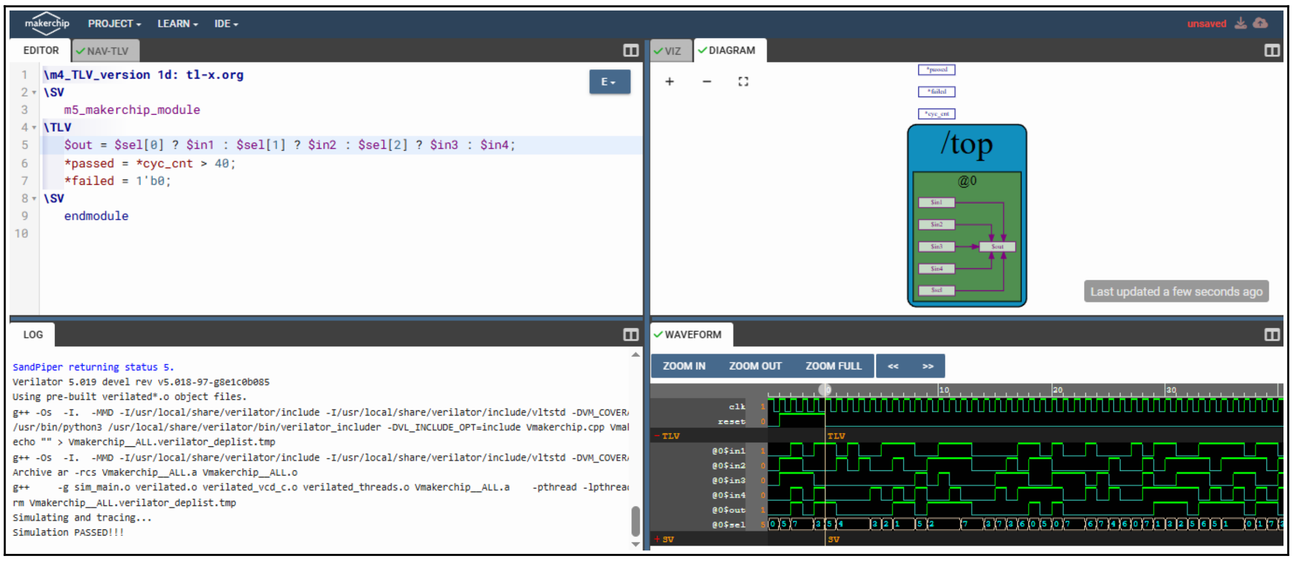


```
\m5_TLV_version 1d: tl-x.org
\m5
\SV
   m5_makerchip_module
\TLV
   $out = $sel ? $in1 : $in2;
\SV
   endmodule
```

Makerchip : https://makerchip.com/sandbox/073fmhN5r/0Mjhqxm

3. 4X1 MUX :

```
\m4_TLV_version 1d: tl-x.org
\SV
   m5_makerchip_module
\TLV
   $out = $sel[0] ? $in1 : $sel[1] ? $in2 : $sel[2] ? $in3 : $in4;
   *passed = *cyc_cnt > 40;
   *failed = 1'b0;
\SV
   endmodule
```



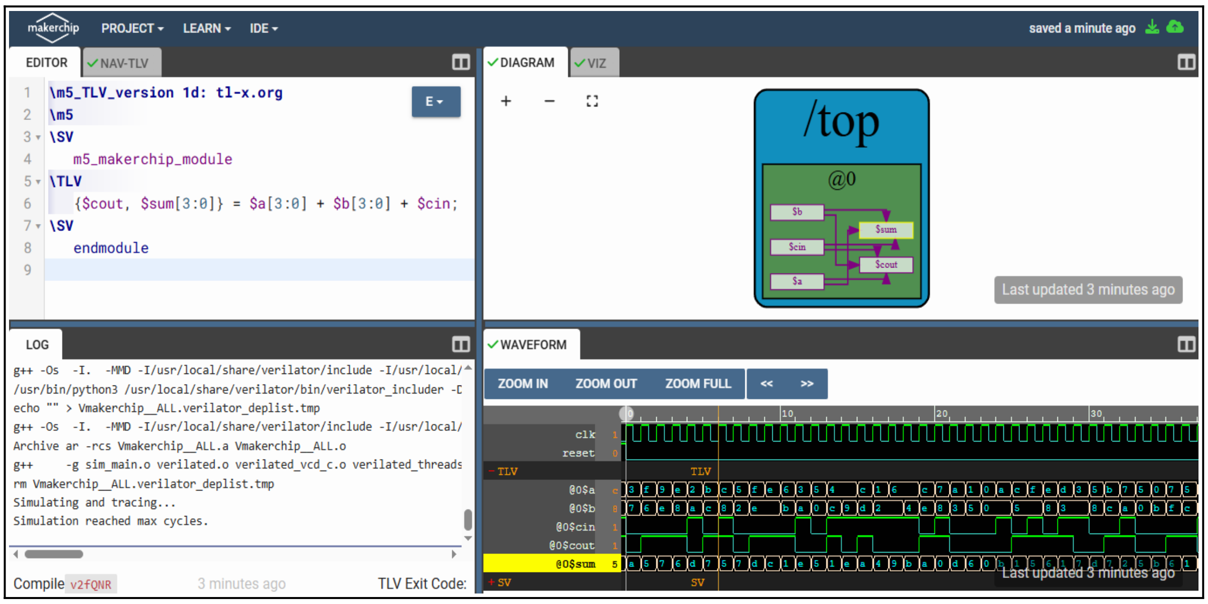
```
\m4_TLV_version 1d: tl-x.org
\SV
   m5_makerchip_module
\TLV
   $out = $sel[0] ? $in1 : $sel[1] ? $in2 : $sel[2] ? $in3 : $in4;
   *passed = *cyc_cnt > 40;
   *failed = 1'b0;
\SV
   endmodule
```

```
SandPiper returning status 5.
Verilator 5.019 devel rev v5.018-97-g8e1c0b085
Using pre-built verilated*.o object files.
g++ -Os  -I.  -MMD -I/usr/local/share/verilator/include -I/usr/local/share/verilator/include/vltstd -DVM_COVER
/usr/bin/python3 /usr/local/share/verilator/bin/verilator_includer -DVL_INCLUDE_OPT=include Vmakerchip.cpp Vmal
echo "" > Vmakerchip__ALL.verilator_deplist.tmp
g++ -Os  -I.  -MMD -I/usr/local/share/verilator/include -I/usr/local/share/verilator/include/vltstd -DVM_COVER
Archive ar -rcs Vmakerchip__ALL.a Vmakerchip__ALL.o
g++      -g sim_main.o verilated.o verilated_vcd_c.o verilated_threads.o Vmakerchip__ALL.a    -pthread -lpthrea
rm Vmakerchip__ALL.verilator_deplist.tmp
Simulating and tracing...
Simulation PASSED!!!
```

Makerchip : https://makerchip.com/sandbox/0rkfAhy2Z/08qh6wO

4. Ripple Carry Adder :

```
\m5_TLV_version 1d: tl-x.org
\m5
\SV
   m5_makerchip_module
\TLV
   {$cout, $sum[3:0]} = $a[3:0] + $b[3:0] + $cin;
\SV
   endmodule
```



```
\m5_TLV_version 1d: tl-x.org
\m5
\SV
   m5_makerchip_module
\TLV
   {$cout, $sum[3:0]} = $a[3:0] + $b[3:0] + $cin;
\SV
   endmodule
```

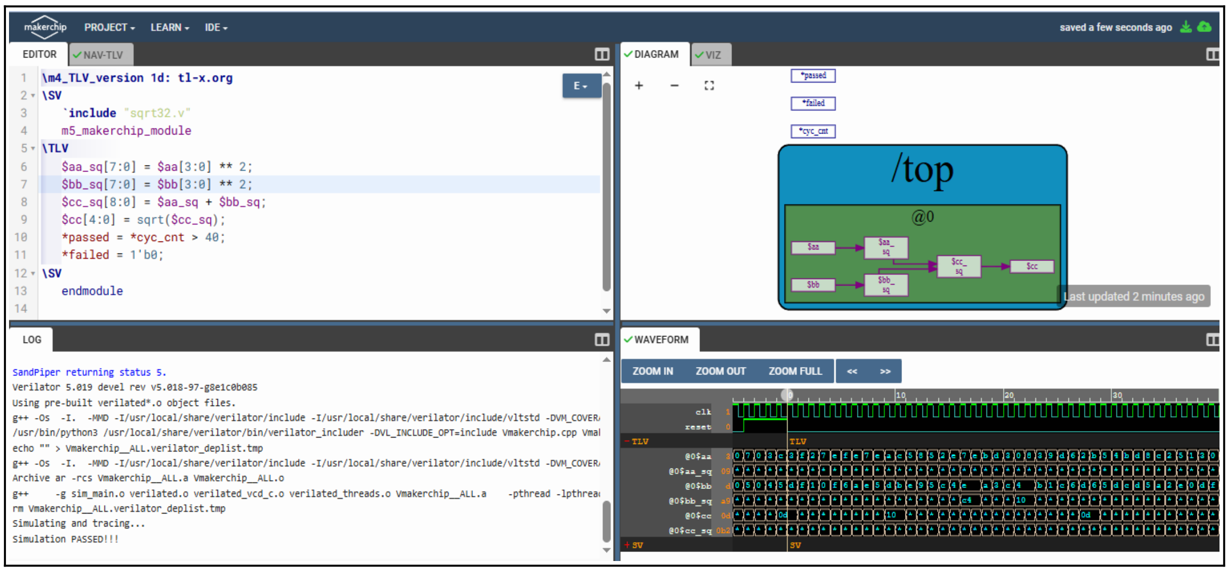
```
g++ -Os  -I.  -MMD -I/usr/local/share/verilator/include -I/usr/local/
/usr/bin/python3 /usr/local/share/verilator/bin/verilator_includer -D
echo "" > Vmakerchip__ALL.verilator_deplist.tmp
g++ -Os  -I.  -MMD -I/usr/local/share/verilator/include -I/usr/local/
Archive ar -rcs Vmakerchip__ALL.a Vmakerchip__ALL.o
g++      -g sim_main.o verilated.o verilated_vcd_c.o verilated_threads
rm Vmakerchip__ALL.verilator_deplist.tmp
Simulating and tracing...
Simulation reached max cycles.
```

Compile v2fQNR 3 minutes ago TLV Exit Code:

Makerchip : https://makerchip.com/sandbox/073fmhN5r/0Nxh0Vm

5. Combinational Pythogorean Theorm :

```
\m4_TLV_version 1d: tl-x.org
\SV
   `include "sqrt32.v"
   m5_makerchip_module
\TLV
   $aa_sq[7:0] = $aa[3:0] ** 2;
   $bb_sq[7:0] = $bb[3:0] ** 2;
   $cc_sq[8:0] = $aa_sq + $bb_sq;
   $cc[4:0] = sqrt($cc_sq);
   *passed = *cyc_cnt > 40;
   *failed = 1'b0;
\SV
   endmodule
```



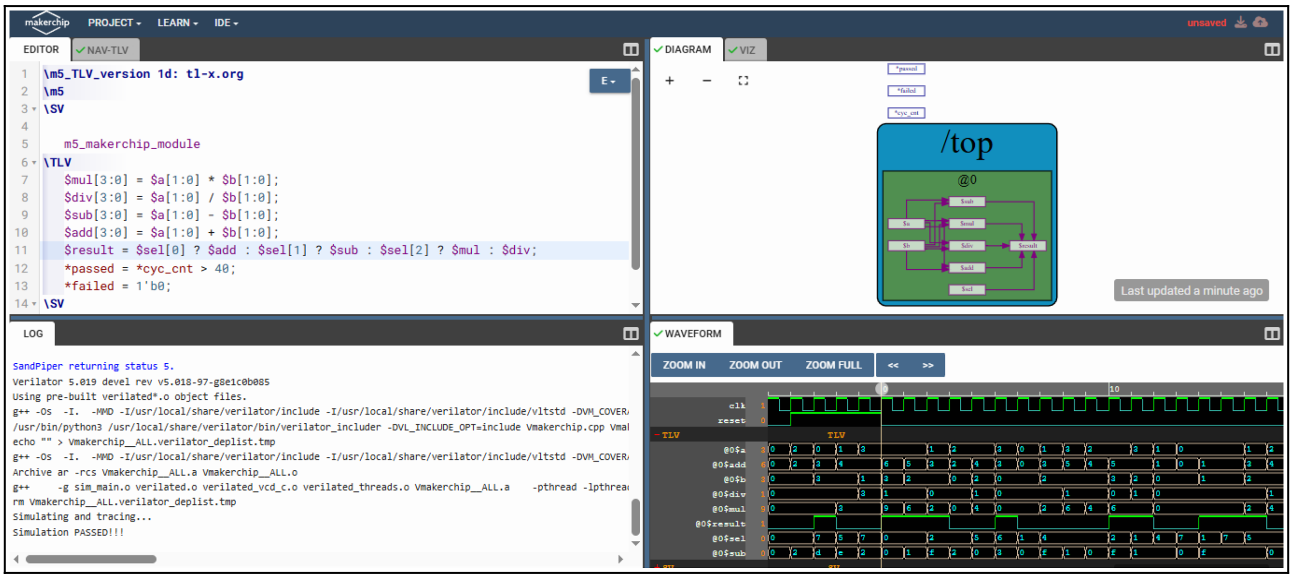
```
\m4_TLV_version 1d: tl-x.org
\SV
   `include "sqrt32.v"
   m5_makerchip_module
\TLV
   $aa_sq[7:0] = $aa[3:0] ** 2;
   $bb_sq[7:0] = $bb[3:0] ** 2;
   $cc_sq[8:0] = $aa_sq + $bb_sq;
   $cc[4:0] = sqrt($cc_sq);
   *passed = *cyc_cnt > 40;
   *failed = 1'b0;
\SV
   endmodule
```

```
SandPiper returning status 5.
Verilator 5.019 devel rev v5.018-97-g8e1c0b085
Using pre-built verilated*.o object files.
g++ -Os  -I.  -MMD -I/usr/local/share/verilator/include -I/usr/local/share/verilator/include/vltstd -DVM_COVER
/usr/bin/python3 /usr/local/share/verilator/bin/verilator_includer -DVL_INCLUDE_OPT=include Vmakerchip.cpp Vmal
echo "" > Vmakerchip__ALL.verilator_deplist.tmp
g++ -Os  -I.  -MMD -I/usr/local/share/verilator/include -I/usr/local/share/verilator/include/vltstd -DVM_COVER
Archive ar -rcs Vmakerchip__ALL.a Vmakerchip__ALL.o
g++     -g sim_main.o verilated.o verilated_vcd_c.o verilated_threads.o Vmakerchip__ALL.a    -pthread -lpthrea
rm Vmakerchip__ALL.verilator_deplist.tmp
Simulating and tracing...
Simulation PASSED!!!
```

Makerchip : https://makerchip.com/sandbox/0rkfAhy2Z/076hAWz

6. Combinational Calculator :

```
\m5_TLV_version 1d: tl-x.org
\m5
\SV
   m5_makerchip_module
\TLV
   $mul[3:0] = $a[1:0] * $b[1:0];
   $div[3:0] = $a[1:0] / $b[1:0];
   $sub[3:0] = $a[1:0] - $b[1:0];
   $add[3:0] = $a[1:0] + $b[1:0];
   $result = $sel[0] ? $add : $sel[1] ? $sub : $sel[2] ? $mul : $div;
   *passed = *cyc_cnt > 40;
   *failed = 1'b0;
\SV
   endmodule
```



```
\m5_TLV_version 1d: tl-x.org
\m5
\SV

   m5_makerchip_module
\TLV
   $mul[3:0] = $a[1:0] * $b[1:0];
   $div[3:0] = $a[1:0] / $b[1:0];
   $sub[3:0] = $a[1:0] - $b[1:0];
   $add[3:0] = $a[1:0] + $b[1:0];
   $result = $sel[0] ? $add : $sel[1] ? $sub : $sel[2] ? $mul : $div;
   *passed = *cyc_cnt > 40;
   *failed = 1'b0;
\SV
```

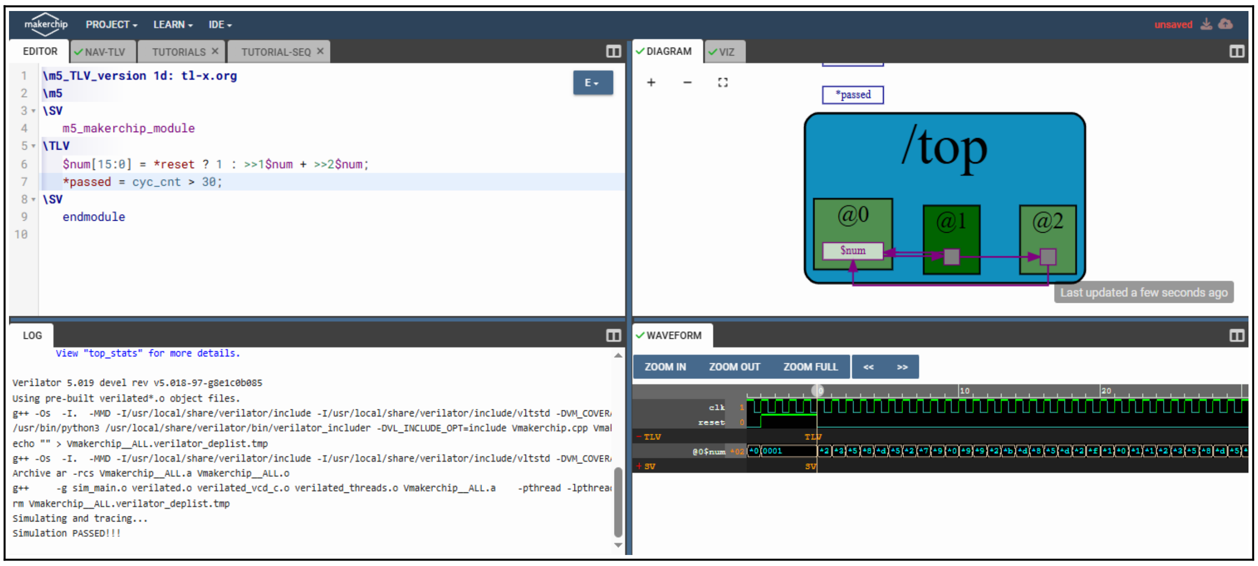
```
SandPiper returning status 5.
Verilator 5.019 devel rev v5.018-97-g8e1c0b085
Using pre-built verilated*.o object files.
g++ -Os  -I.  -MMD -I/usr/local/share/verilator/include -I/usr/local/share/verilator/include/vltstd -DVM_COVER
/usr/bin/python3 /usr/local/share/verilator/bin/verilator_includer -DVL_INCLUDE_OPT=include Vmakerchip.cpp Vmal
echo "" > Vmakerchip__ALL.verilator_deplist.tmp
g++ -Os  -I.  -MMD -I/usr/local/share/verilator/include -I/usr/local/share/verilator/include/vltstd -DVM_COVER
Archive ar -rcs Vmakerchip__ALL.a Vmakerchip__ALL.o
g++     -g sim_main.o verilated.o verilated_vcd_c.o verilated_threads.o Vmakerchip__ALL.a    -pthread -lpthrea
rm Vmakerchip__ALL.verilator_deplist.tmp
Simulating and tracing...
Simulation PASSED!!!
```

Makerchip : https://makerchip.com/sandbox/0rkfAhy2Z/098hkYY

Sequential logic in TL-Verilog :

1. Fibinacci Sequence :

```
\m5_TLV_version 1d: tl-x.org
\m5
\SV
   m5_makerchip_module
\TLV
   $num[15:0] = *reset ? 1 : >>1$num + >>2$num;
   *passed = cyc_cnt > 30;
\SV
   endmodule
```



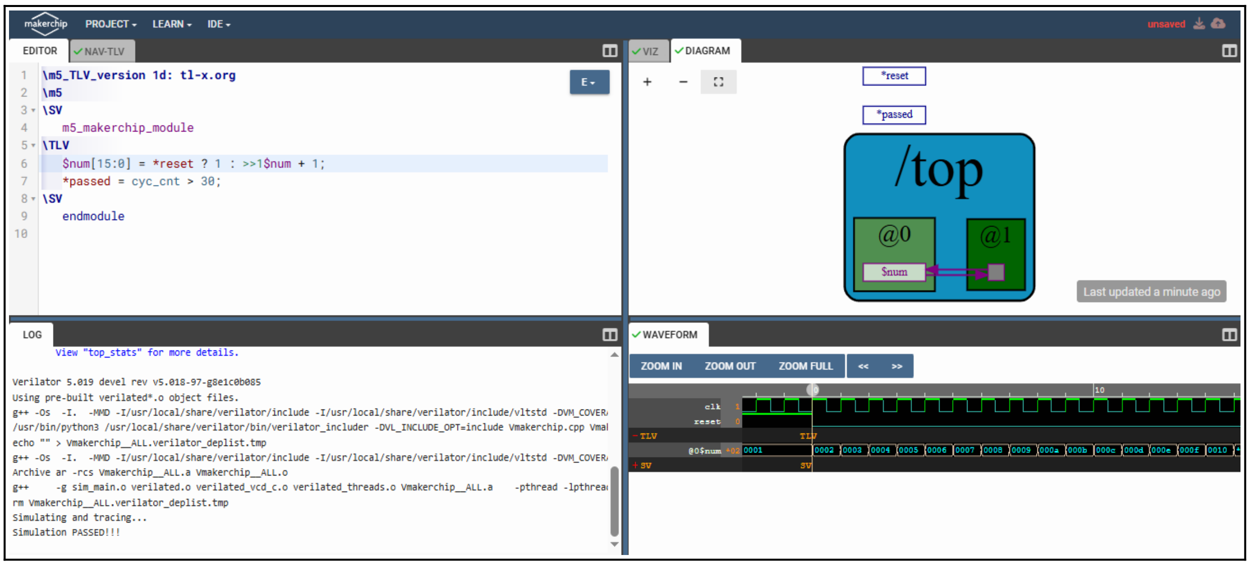
```
\m5_TLV_version 1d: tl-x.org
\m5
\SV
   m5_makerchip_module
\TLV
   $num[15:0] = *reset ? 1 : >>1$num + >>2$num;
   *passed = cyc_cnt > 30;
\SV
   endmodule
```

```
        View "top_stats" for more details.

Verilator 5.019 devel rev v5.018-97-g8e1c0b085
Using pre-built verilated*.o object files.
g++ -Os  -I.  -MMD -I/usr/local/share/verilator/include -I/usr/local/share/verilator/include/vltstd -DVM_COVER
/usr/bin/python3 /usr/local/share/verilator/bin/verilator_includer -DVL_INCLUDE_OPT=include Vmakerchip.cpp Vmal
echo "" > Vmakerchip__ALL.verilator_deplist.tmp
g++ -Os  -I.  -MMD -I/usr/local/share/verilator/include -I/usr/local/share/verilator/include/vltstd -DVM_COVER
Archive ar -rcs Vmakerchip__ALL.a Vmakerchip__ALL.o
g++     -g sim_main.o verilated.o verilated_vcd_c.o verilated_threads.o Vmakerchip__ALL.a    -pthread -lpthrea
rm Vmakerchip__ALL.verilator_deplist.tmp
Simulating and tracing...
Simulation PASSED!!!
```

Makerchip : https://makerchip.com/sandbox/0rkfAhy2Z/00ghGrm

2. Counter :

```
\m5_TLV_version 1d: tl-x.org
\m5
\SV
   m5_makerchip_module
\TLV
   $num[15:0] = *reset ? 1 : >>1$num + 1;
   *passed = cyc_cnt > 30;
\SV
   endmodule
```



```
\m5_TLV_version 1d: tl-x.org
\m5
\SV
   m5_makerchip_module
\TLV
   $num[15:0] = *reset ? 1 : >>1$num + 1;
   *passed = cyc_cnt > 30;
\SV
   endmodule
```

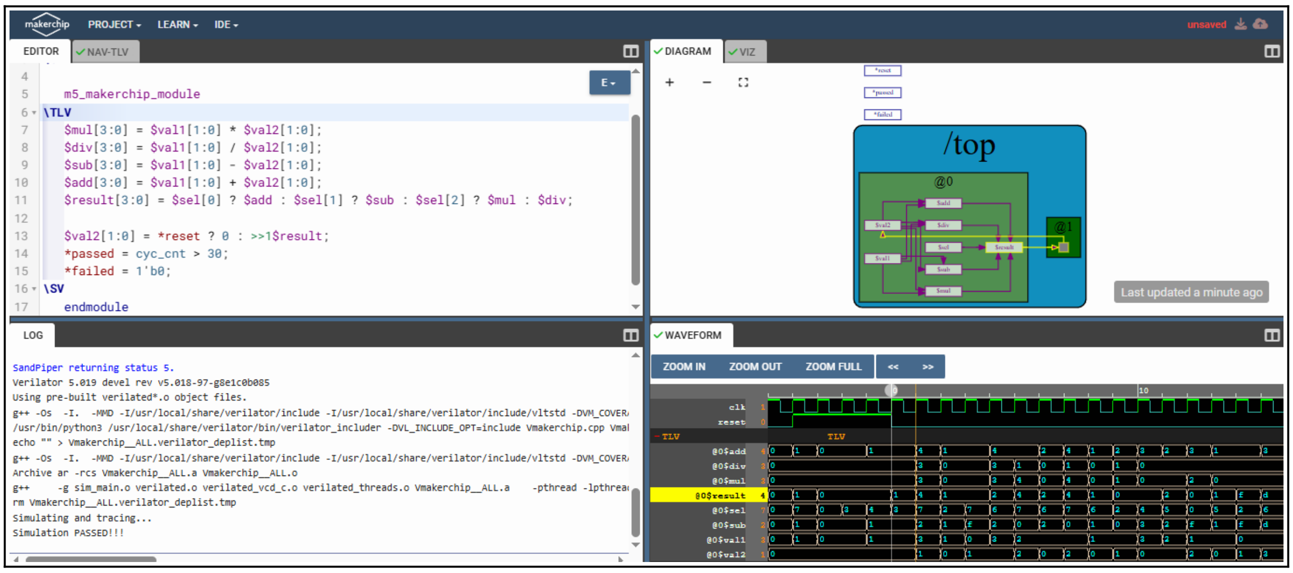
```
      View "top_stats" for more details.

Verilator 5.019 devel rev v5.018-97-g8e1c0b085
Using pre-built verilated*.o object files.
g++ -Os  -I.  -MMD -I/usr/local/share/verilator/include -I/usr/local/share/verilator/include/vltstd -DVM_COVER
/usr/bin/python3 /usr/local/share/verilator/bin/verilator_includer -DVL_INCLUDE_OPT=include Vmakerchip.cpp Vmal
echo "" > Vmakerchip__ALL.verilator_deplist.tmp
g++ -Os  -I.  -MMD -I/usr/local/share/verilator/include -I/usr/local/share/verilator/include/vltstd -DVM_COVER
Archive ar -rcs Vmakerchip__ALL.a Vmakerchip__ALL.o
g++     -g sim_main.o verilated.o verilated_vcd_c.o verilated_threads.o Vmakerchip__ALL.a    -pthread -lpthrea
rm Vmakerchip__ALL.verilator_deplist.tmp
Simulating and tracing...
Simulation PASSED!!!
```

```
clk     1
reset   0
- TLV
@0$num  +02  0001 0002 0003 0004 0005 0006 0007 0008 0009 000a 000b 000c 000d 000e 000f 0010
+ SV
```

Makerchip : https://makerchip.com/sandbox/0rkfAhy2Z/0g5hAYw

3. Sequential Calculator :

```
\m5_TLV_version 1d: tl-x.org
\m5
\SV

   m5_makerchip_module
\TLV
   $mul[3:0] = $val1[1:0] * $val2[1:0];
   $div[3:0] = $val1[1:0] / $val2[1:0];
   $sub[3:0] = $val1[1:0] - $val2[1:0];
   $add[3:0] = $val1[1:0] + $val2[1:0];
   $result[3:0] = $sel[0] ? $add : $sel[1] ? $sub : $sel[2] ? $mul : $div;

   $val2[1:0] = *reset ? 0 : >>1$result;
   *passed = cyc_cnt > 30;
   *failed = 1'b0;
\SV
   endmodule
```



```
   m5_makerchip_module
\TLV
   $mul[3:0] = $val1[1:0] * $val2[1:0];
   $div[3:0] = $val1[1:0] / $val2[1:0];
   $sub[3:0] = $val1[1:0] - $val2[1:0];
   $add[3:0] = $val1[1:0] + $val2[1:0];
   $result[3:0] = $sel[0] ? $add : $sel[1] ? $sub : $sel[2] ? $mul : $div;

   $val2[1:0] = *reset ? 0 : >>1$result;
   *passed = cyc_cnt > 30;
   *failed = 1'b0;
\SV
   endmodule
```

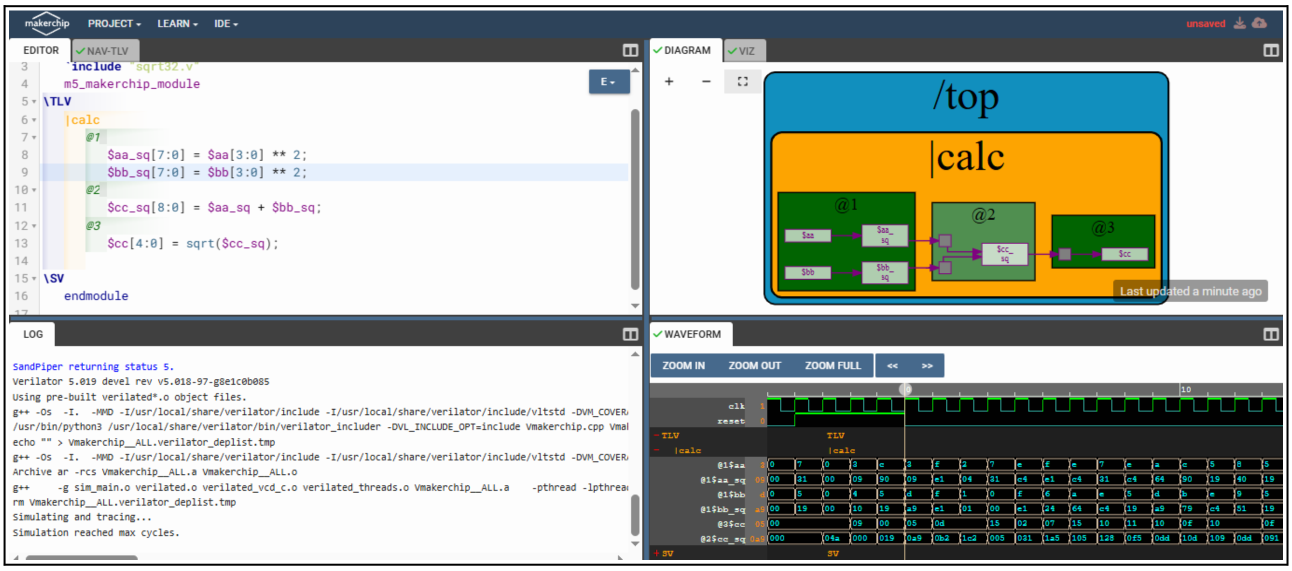
```
SandPiper returning status 5.
Verilator 5.019 devel rev v5.018-97-g8e1c0b085
Using pre-built verilated*.o object files.
g++ -Os  -I.  -MMD -I/usr/local/share/verilator/include -I/usr/local/share/verilator/include/vltstd -DVM_COVER
/usr/bin/python3 /usr/local/share/verilator/bin/verilator_includer -DVL_INCLUDE_OPT=include Vmakerchip.cpp Vmal
echo "" > Vmakerchip__ALL.verilator_deplist.tmp
g++ -Os  -I.  -MMD -I/usr/local/share/verilator/include -I/usr/local/share/verilator/include/vltstd -DVM_COVER
Archive ar -rcs Vmakerchip__ALL.a Vmakerchip__ALL.o
g++     -g sim_main.o verilated.o verilated_vcd_c.o verilated_threads.o Vmakerchip__ALL.a    -pthread -lpthrea
rm Vmakerchip__ALL.verilator_deplist.tmp
Simulating and tracing...
Simulation PASSED!!!
```

Makerchip : https://makerchip.com/sandbox/0rkfAhy2Z/0k5hOq4

Pipelining Logic in TL-Verilog :

1. Pipelined Pythogorean Theorm :

```
\m4_TLV_version 1d: tl-x.org
\SV
        `include "sqrt32.v"
   m5_makerchip_module
\TLV
   |calc
      @1
         $aa_sq[7:0] = $aa[3:0] ** 2;
         $bb_sq[7:0] = $bb[3:0] ** 2;
      @2
         $cc_sq[8:0] = $aa_sq + $bb_sq;
      @3
         $cc[4:0] = sqrt($cc_sq);
\SV
   endmodule
```



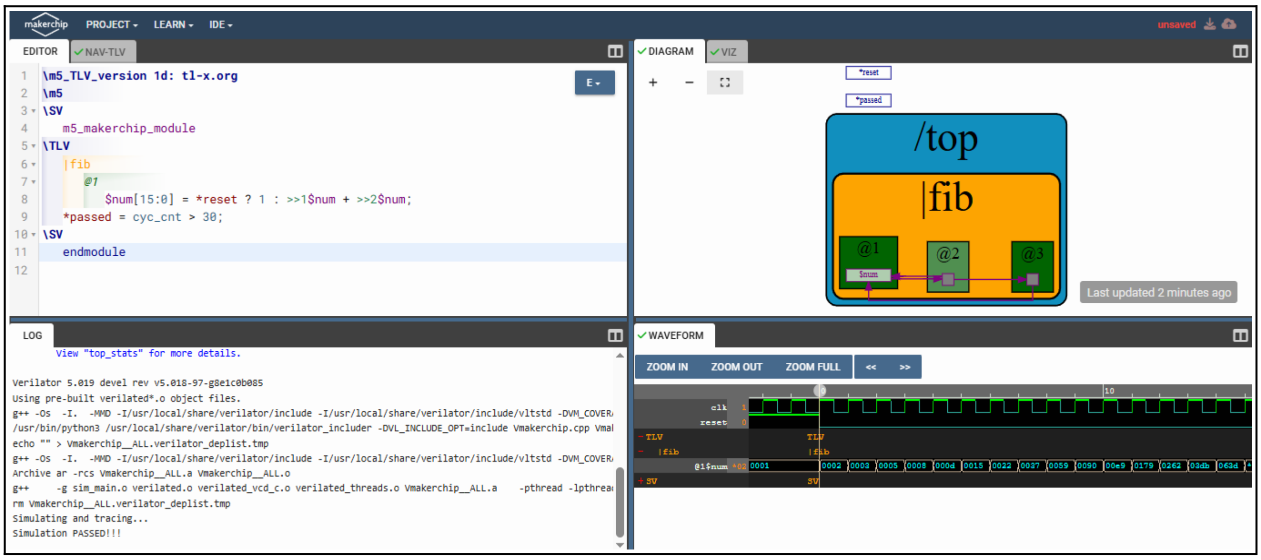
```
   `include "sqrt32.v"
   m5_makerchip_module
\TLV
   |calc
      @1
         $aa_sq[7:0] = $aa[3:0] ** 2;
         $bb_sq[7:0] = $bb[3:0] ** 2;
      @2
         $cc_sq[8:0] = $aa_sq + $bb_sq;
      @3
         $cc[4:0] = sqrt($cc_sq);

\SV
   endmodule
```

```
SandPiper returning status 5.
Verilator 5.019 devel rev v5.018-97-g8e1c0b085
Using pre-built verilated*.o object files.
g++ -Os  -I.  -MMD -I/usr/local/share/verilator/include -I/usr/local/share/verilator/include/vltstd -DVM_COVER
/usr/bin/python3 /usr/local/share/verilator/bin/verilator_includer -DVL_INCLUDE_OPT=include Vmakerchip.cpp Vmal
echo "" > Vmakerchip__ALL.verilator_deplist.tmp
g++ -Os  -I.  -MMD -I/usr/local/share/verilator/include -I/usr/local/share/verilator/include/vltstd -DVM_COVER
Archive ar -rcs Vmakerchip__ALL.a Vmakerchip__ALL.o
g++     -g sim_main.o verilated.o verilated_vcd_c.o verilated_threads.o Vmakerchip__ALL.a    -pthread -lpthrea
rm Vmakerchip__ALL.verilator_deplist.tmp
Simulating and tracing...
Simulation reached max cycles.
```

Makerchip : https://makerchip.com/sandbox/0rkfAhy2Z/0lOh2z6

2. Pipelined Fibonacci Sequence :

```
\m5_TLV_version 1d: tl-x.org
\m5
\SV
   m5_makerchip_module
\TLV
   |fib
      @1
         $num[15:0] = *reset ? 1 : >>1$num + >>2$num;
   *passed = cyc_cnt > 30;
\SV
   endmodule
```



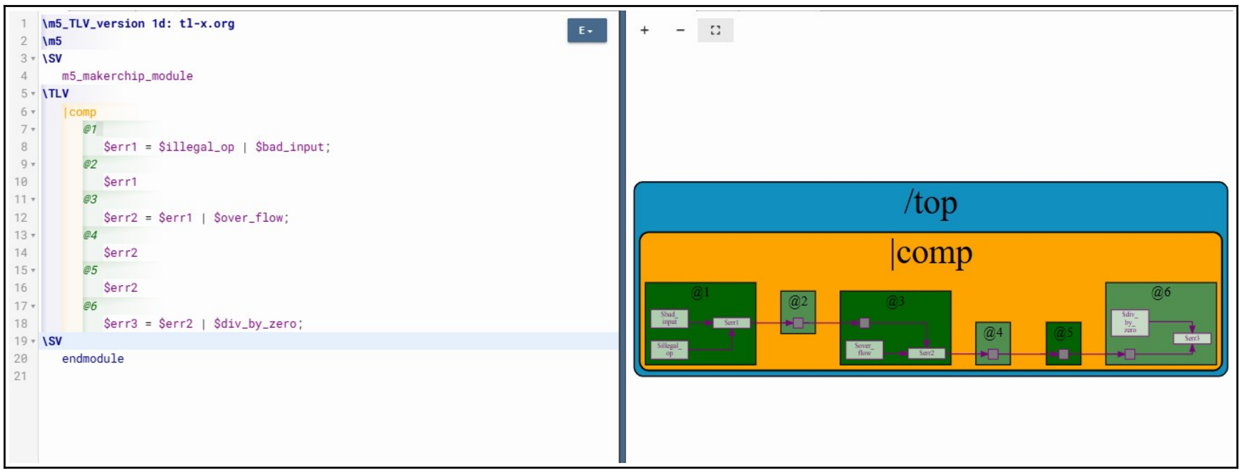
```
\m5_TLV_version 1d: tl-x.org
\m5
\SV
   m5_makerchip_module
\TLV
   |fib
      @1
         $num[15:0] = *reset ? 1 : >>1$num + >>2$num;
   *passed = cyc_cnt > 30;
\SV
   endmodule
```

```
      View "top_stats" for more details.

Verilator 5.019 devel rev v5.018-97-g8e1c0b085
Using pre-built verilated*.o object files.
g++ -Os  -I.  -MMD -I/usr/local/share/verilator/include -I/usr/local/share/verilator/include/vltstd -DVM_COVER
/usr/bin/python3 /usr/local/share/verilator/bin/verilator_includer -DVL_INCLUDE_OPT=include Vmakerchip.cpp Vmal
echo "" > Vmakerchip__ALL.verilator_deplist.tmp
g++ -Os  -I.  -MMD -I/usr/local/share/verilator/include -I/usr/local/share/verilator/include/vltstd -DVM_COVER
Archive ar -rcs Vmakerchip__ALL.a Vmakerchip__ALL.o
g++     -g sim_main.o verilated.o verilated_vcd_c.o verilated_threads.o Vmakerchip__ALL.a    -pthread -lpthrea
rm Vmakerchip__ALL.verilator_deplist.tmp
Simulating and tracing...
Simulation PASSED!!!
```

Makerchip : https://makerchip.com/sandbox/0rkfAhy2Z/0mwhjR8

3. Error Conditions within Computational Pipeline :

```
\m5_TLV_version 1d: tl-x.org
\m5
\SV
        m5_makerchip_module
\TLV
  |comp
    @1
      $err1 = $illegal_op | $bad_input;
    @2
      $err1
    @3
      $err2 = $err1 | $over_flow;
    @4
      $err2
    @5
      $err2
    @6
      $err3 = $err2 | $div_by_zero;
\SV
  endmodule
```

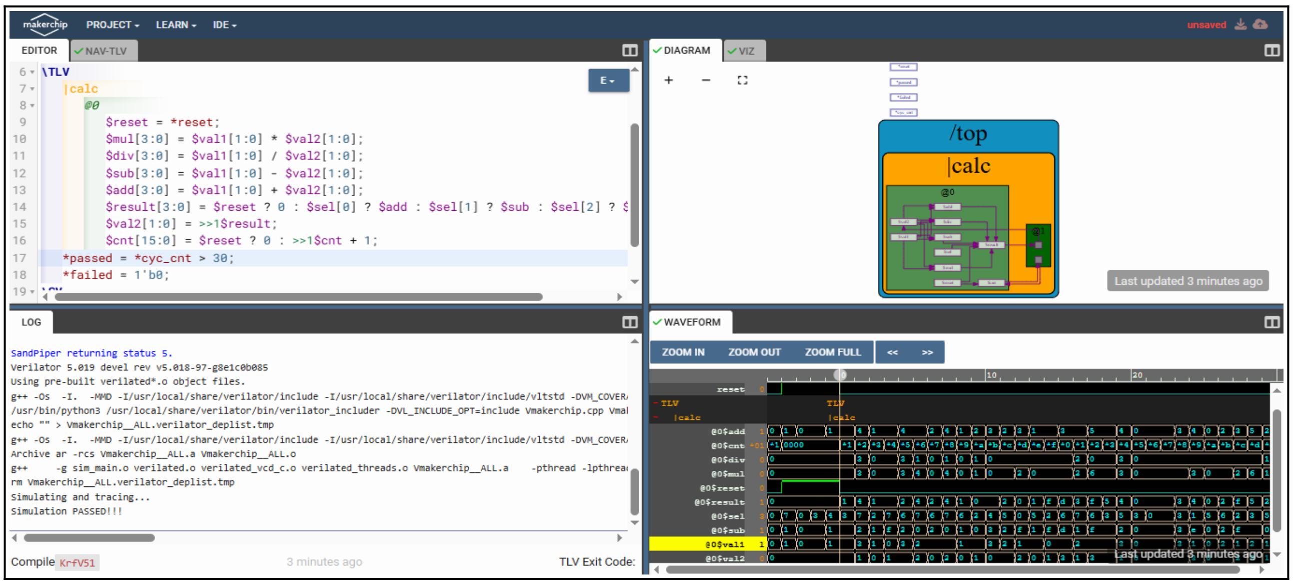


```
\m5_TLV_version 1d: tl-x.org
\m5
\SV
   m5_makerchip_module
\TLV
   |comp
      @1
         $err1 = $illegal_op | $bad_input;
      @2
         $err1
      @3
         $err2 = $err1 | $over_flow;
      @4
         $err2
      @5
         $err2
      @6
         $err3 = $err2 | $div_by_zero;
\SV
   endmodule
```

Makerchip : https://makerchip.com/sandbox/0rkfAhy2Z/0nZh76n

4. Single Cycle Counter Pipeline for Sequential Calculator :

```
\m5_TLV_version 1d: tl-x.org
\m5
\SV
   m5_makerchip_module
\TLV
   |calc
      @0
         $reset = *reset;
         $mul[3:0] = $val1[1:0] * $val2[1:0];
         $div[3:0] = $val1[1:0] / $val2[1:0];
         $sub[3:0] = $val1[1:0] - $val2[1:0];
         $add[3:0] = $val1[1:0] + $val2[1:0];
         $result[3:0] = $reset ? 0 : $sel[0] ? $add : $sel[1] ? $sub : $sel[2] ? $mul : $div;
         $val2[1:0] = >>1$result;
         $cnt[15:0] = $reset ? 0 : >>1$cnt + 1;
   *passed = *cyc_cnt > 30;
   *failed = 1'b0;
\SV
   endmodule
```



```
\TLV
   |calc
      @0
         $reset = *reset;
         $mul[3:0] = $val1[1:0] * $val2[1:0];
         $div[3:0] = $val1[1:0] / $val2[1:0];
         $sub[3:0] = $val1[1:0] - $val2[1:0];
         $add[3:0] = $val1[1:0] + $val2[1:0];
         $result[3:0] = $reset ? 0 : $sel[0] ? $add : $sel[1] ? $sub : $sel[2] ? $
         $val2[1:0] = >>1$result;
         $cnt[15:0] = $reset ? 0 : >>1$cnt + 1;
   *passed = *cyc_cnt > 30;
   *failed = 1'b0;
```

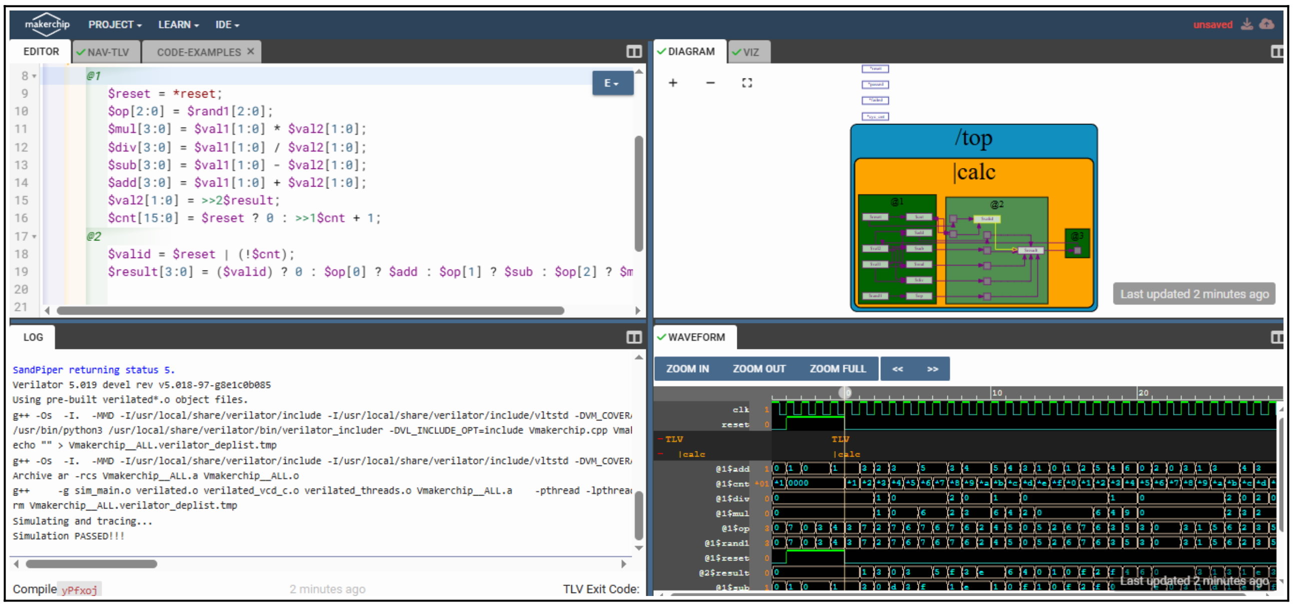
```
SandPiper returning status 5.
Verilator 5.019 devel rev v5.018-97-g8e1c0b085
Using pre-built verilated*.o object files.
g++ -Os  -I.  -MMD -I/usr/local/share/verilator/include -I/usr/local/share/verilator/include/vltstd -DVM_COVER
/usr/bin/python3 /usr/local/share/verilator/bin/verilator_includer -DVL_INCLUDE_OPT=include Vmakerchip.cpp Vmal
echo "" > Vmakerchip__ALL.verilator_deplist.tmp
g++ -Os  -I.  -MMD -I/usr/local/share/verilator/include -I/usr/local/share/verilator/include/vltstd -DVM_COVER
Archive ar -rcs Vmakerchip__ALL.a Vmakerchip__ALL.o
g++     -g sim_main.o verilated.o verilated_vcd_c.o verilated_threads.o Vmakerchip__ALL.a    -pthread -lpthrea
rm Vmakerchip__ALL.verilator_deplist.tmp
Simulating and tracing...
Simulation PASSED!!!
```

Compile KrfV51   3 minutes ago   TLV Exit Code:

Makerchip : https://makerchip.com/sandbox/0rkfAhy2Z/0oYhrKJ

5. 2-cycle pipelined Calculator :

```
\m5_TLV_version 1d: tl-x.org
\m5
\SV
   m5_makerchip_module
\TLV
   |calc
      @1
         $reset = *reset;
         $op[2:0] = $rand1[2:0];
         $mul[3:0] = $val1[1:0] * $val2[1:0];
         $div[3:0] = $val1[1:0] / $val2[1:0];
         $sub[3:0] = $val1[1:0] - $val2[1:0];
         $add[3:0] = $val1[1:0] + $val2[1:0];
         $val2[1:0] = >>2$result;
         $cnt[15:0] = $reset ? 0 : >>1$cnt + 1;
      @2
         $valid = $reset | (!$cnt);
         $result[3:0] = ($valid) ? 0 : $op[0] ? $add : $op[1] ? $sub : $op[2] ? $mul : $div;
   *passed = *cyc_cnt > 30;
   *failed = 1'b0;
\SV
   endmodule
```



```
      @1
         $reset = *reset;
         $op[2:0] = $rand1[2:0];
         $mul[3:0] = $val1[1:0] * $val2[1:0];
         $div[3:0] = $val1[1:0] / $val2[1:0];
         $sub[3:0] = $val1[1:0] - $val2[1:0];
         $add[3:0] = $val1[1:0] + $val2[1:0];
         $val2[1:0] = >>2$result;
         $cnt[15:0] = $reset ? 0 : >>1$cnt + 1;
      @2
         $valid = $reset | (!$cnt);
         $result[3:0] = ($valid) ? 0 : $op[0] ? $add : $op[1] ? $sub : $op[2] ? $m
```

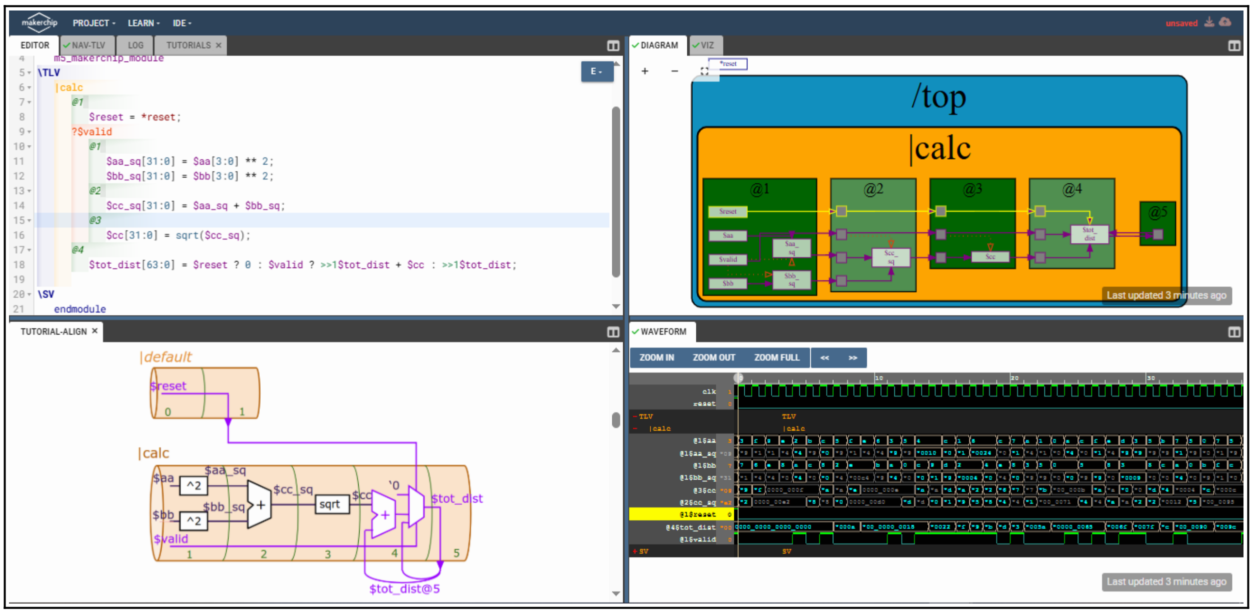
```
SandPiper returning status 5.
Verilator 5.019 devel rev v5.018-97-g8e1c0b085
Using pre-built verilated*.o object files.
g++ -Os  -I.  -MMD -I/usr/local/share/verilator/include -I/usr/local/share/verilator/include/vltstd -DVM_COVER
/usr/bin/python3 /usr/local/share/verilator/bin/verilator_includer -DVL_INCLUDE_OPT=include Vmakerchip.cpp Vmal
echo "" > Vmakerchip__ALL.verilator_deplist.tmp
g++ -Os  -I.  -MMD -I/usr/local/share/verilator/include -I/usr/local/share/verilator/include/vltstd -DVM_COVER
Archive ar -rcs Vmakerchip__ALL.a Vmakerchip__ALL.o
g++     -g sim_main.o verilated.o verilated_vcd_c.o verilated_threads.o Vmakerchip__ALL.a    -pthread -lpthrea
rm Vmakerchip__ALL.verilator_deplist.tmp
Simulating and tracing...
Simulation PASSED!!!
```

Makerchip : https://makerchip.com/sandbox/0rkfAhy2Z/0qjh874

Validity in TL-Verilog :

1. Computation of Total Distance with Validity :

```
\m4_TLV_version 1d: tl-x.org
\SV
   `include "sqrt32.v"
   m5_makerchip_module
\TLV
   |calc
      @1
         $reset = *reset;
      ?$valid
         @1
            $aa_sq[31:0] = $aa[3:0] ** 2;
            $bb_sq[31:0] = $bb[3:0] ** 2;
         @2
            $cc_sq[31:0] = $aa_sq + $bb_sq;
         @3
            $cc[31:0] = sqrt($cc_sq);
      @4
         $tot_dist[63:0] = $reset ? 0 : $valid ? >>1$tot_dist + $cc : >>1$tot_dist;
\SV
   endmodule
```

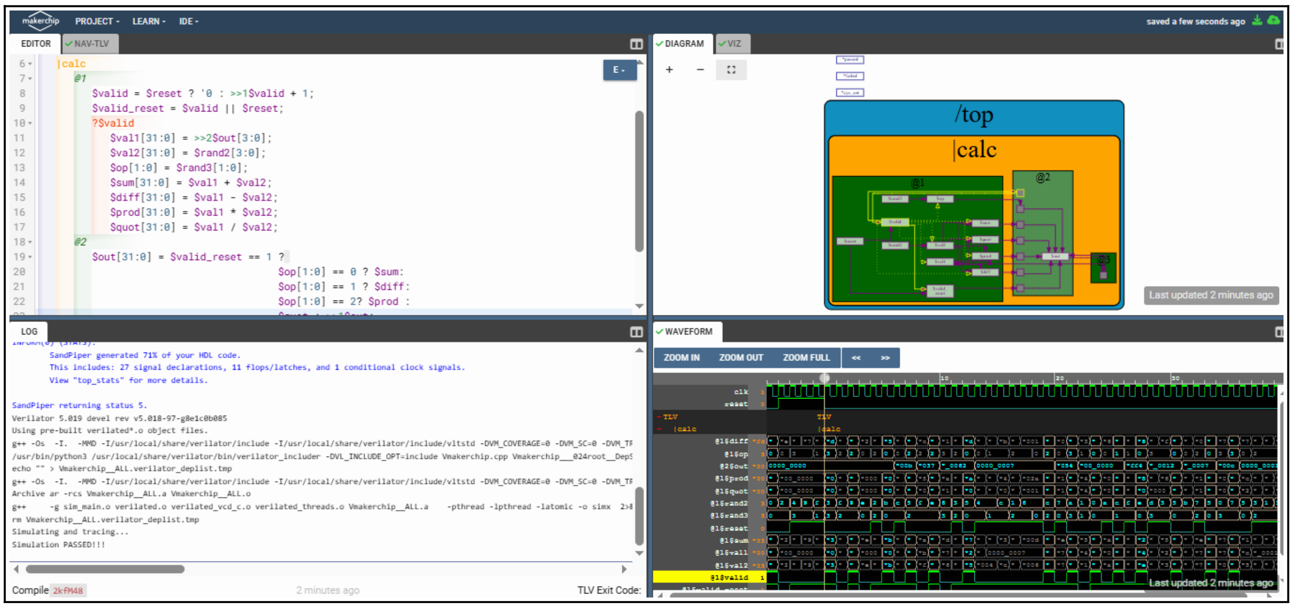


```
   m5_makerchip_module
\TLV
   |calc
      @1
         $reset = *reset;
      ?$valid
         @1
            $aa_sq[31:0] = $aa[3:0] ** 2;
            $bb_sq[31:0] = $bb[3:0] ** 2;
         @2
            $cc_sq[31:0] = $aa_sq + $bb_sq;
         @3
            $cc[31:0] = sqrt($cc_sq);
      @4
         $tot_dist[63:0] = $reset ? 0 : $valid ? >>1$tot_dist + $cc : >>1$tot_dist;
\SV
   endmodule
```

Makerchip : https://makerchip.com/sandbox/073fmhNyx/0vgh7yK

2. 2-cycle Calculator with Validity :

```
\m5_TLV_version 1d: tl-x.org
\m5
\SV
   m5_makerchip_module
\TLV
   |calc
      @1
         $valid = $reset ? '0 : >>1$valid + 1;
         $valid_reset = $valid || $reset;
         ?$valid
            $val1[31:0] = >>2$out[3:0];
            $val2[31:0] = $rand2[3:0];
            $op[1:0] = $rand3[1:0];
            $sum[31:0] = $val1 + $val2;
            $diff[31:0] = $val1 - $val2;
            $prod[31:0] = $val1 * $val2;
            $quot[31:0] = $val1 / $val2;
      @2
         $out[31:0] = $valid_reset == 1 ?
                                    $op[1:0] == 0 ? $sum:
                                    $op[1:0] == 1 ? $diff:
                                    $op[1:0] == 2? $prod :
                                    $quot : >>1$out;
   *passed = *cyc_cnt > 40;
   *failed = 1'b0;
\SV
   endmodule
```



```
   |calc
      @1
         $valid = $reset ? '0 : >>1$valid + 1;
         $valid_reset = $valid || $reset;
         ?$valid
            $val1[31:0] = >>2$out[3:0];
            $val2[31:0] = $rand2[3:0];
            $op[1:0] = $rand3[1:0];
            $sum[31:0] = $val1 + $val2;
            $diff[31:0] = $val1 - $val2;
            $prod[31:0] = $val1 * $val2;
            $quot[31:0] = $val1 / $val2;
      @2
         $out[31:0] = $valid_reset == 1 ?
                                    $op[1:0] == 0 ? $sum:
                                    $op[1:0] == 1 ? $diff:
                                    $op[1:0] == 2? $prod :
```

```
        SandPiper generated 71% of your HDL code.
        This includes: 27 signal declarations, 11 flops/latches, and 1 conditional clock signals.
        View "top_stats" for more details.

SandPiper returning status 5.
Verilator 5.019 devel rev v5.018-97-g8e1c0b085
Using pre-built verilated*.o object files.
g++ -Os  -I.  -MMD -I/usr/local/share/verilator/include -I/usr/local/share/verilator/include/vltstd -DVM_COVERAGE=0 -DVM_SC=0 -DVM_TR
/usr/bin/python3 /usr/local/share/verilator/bin/verilator_includer -DVL_INCLUDE_OPT=include Vmakerchip.cpp Vmakerchip___024root__DepS
echo "" > Vmakerchip__ALL.verilator_deplist.tmp
g++ -Os  -I.  -MMD -I/usr/local/share/verilator/include -I/usr/local/share/verilator/include/vltstd -DVM_COVERAGE=0 -DVM_SC=0 -DVM_TR
Archive ar -rcs Vmakerchip__ALL.a Vmakerchip__ALL.o
g++     -g sim_main.o verilated.o verilated_vcd_c.o verilated_threads.o Vmakerchip__ALL.a    -pthread -lpthread -latomic -o simx  2>&
rm Vmakerchip__ALL.verilator_deplist.tmp
Simulating and tracing...
Simulation PASSED!!!
```

Makerchip : https://makerchip.com/sandbox/073fmhNyx/0AnhN18