



# Simtera 4.0 Кроссплатформенная и российская

Мальшев Никита

[www.eremex.ru](http://www.eremex.ru)



Компания разрабатывает САПР  
электроники с 2007 года



Первая российская САПР РЭА,  
обеспечивающая сквозной цикл  
проектирования электроники





Госкорпорация «Роскосмос»

**ЦКБ ГЕОФИЗИКА**

Замена P-CAD различных версий

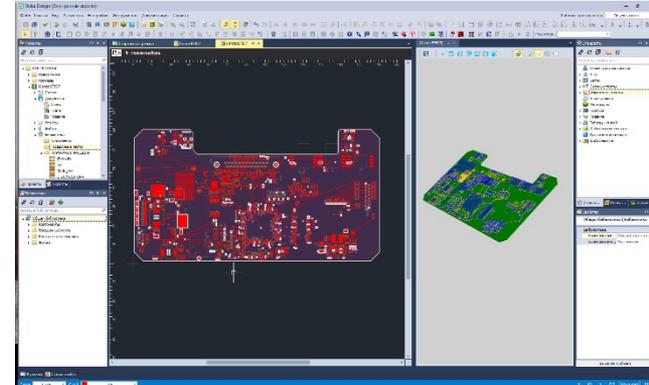
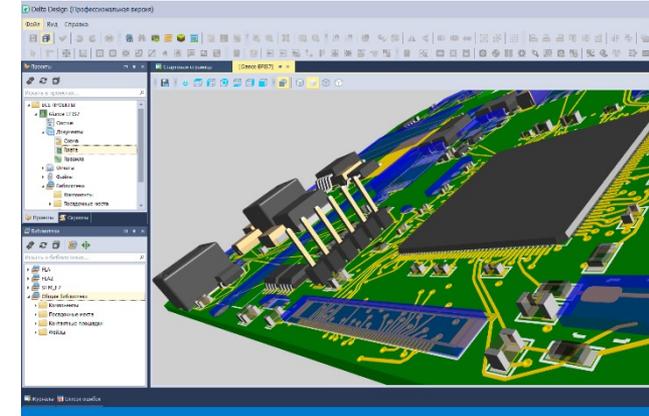
## Разработка изделия

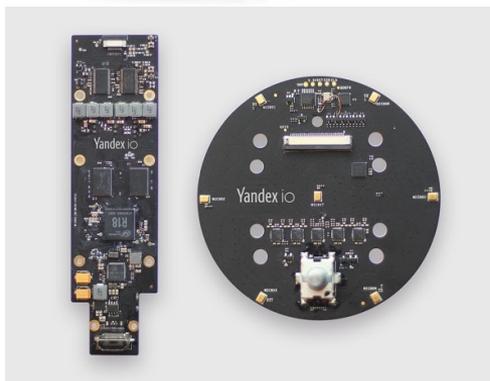
- Модернизация рабочих мест и замена устаревшего P-CAD на Delta Design;
- Проектирование механической части устройства КОМПАС-3D;
- Проектирование печатной платы в Delta Design;
  - Схемотехническое проектирование;
  - Проектирование конструкции и топологии;
  - Подготовка производства.



# GLANCE AVIONICS

Пилотажно-навигационные приборы Glance EFIS - высокопроизводительное решение для любых типов воздушных судов.



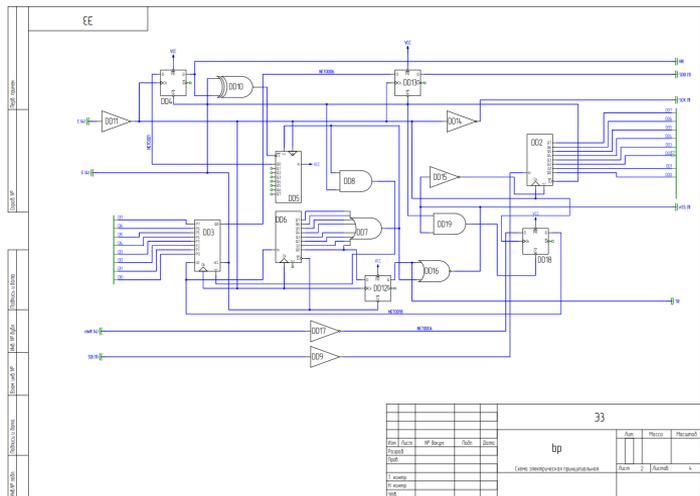


# Яндекс

1. Удобство трассировки
2. Скорость трассировки в разы выше, чем в других программах
3. Возможность уложиться в меньшее кол-во слоёв
4. Использование бессеточной трассировки в узких местах позволяет оптимизировать класс точности платы

# Разработка высокоуровневого описания

Раньше:



**Схемотехнический редактор:**

*Работа с компонентами, цепями, шинами, текстом и пр.*

*Поддержка многolistовых схем, блоков*

*Штамп по ГОСТ*

**Текстовый редактор:**

*Подсветка синтаксиса*

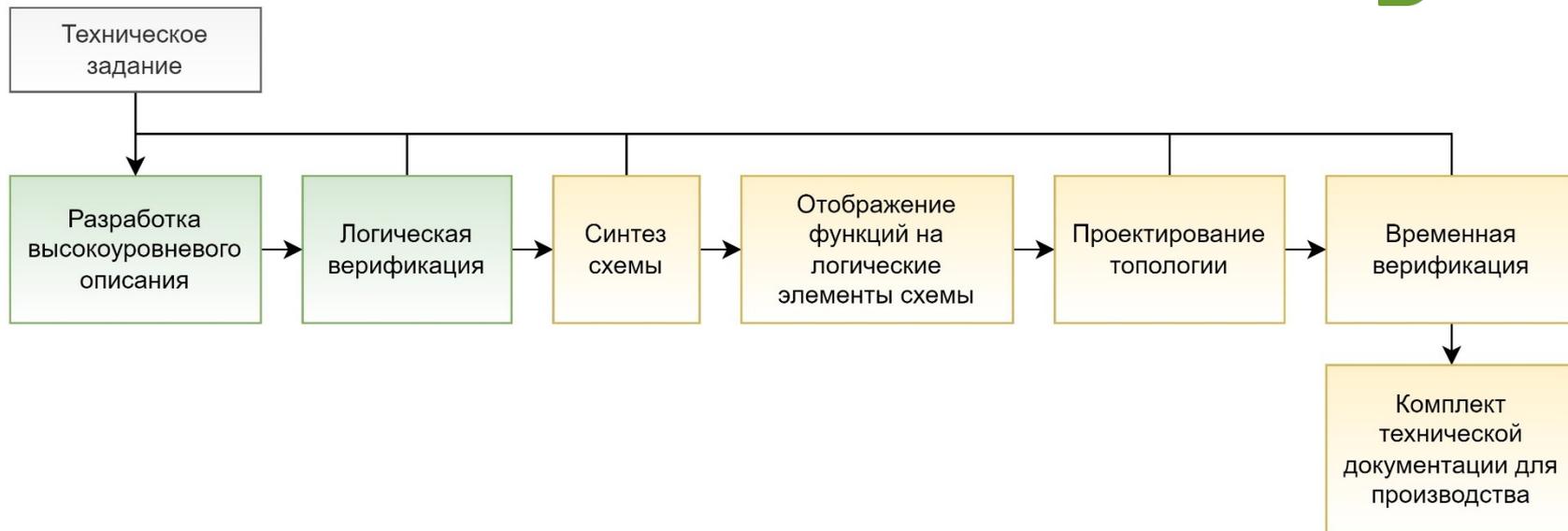
*Поиск по тексту, по проекту, по регулярным выражениям*

*Авто-дополнение кода и подсказки*

Сейчас:

```
1 ////////////////////////////////////////////////////////////////////
2 // File name       : SPIMaster.sv
3 // Author          : Nikita Malyshev
4 // Email           : malyshev.n@eremex.ru
5 // Phone          :
6 // Company         : Eremex
7 // Dep             :
8 // Description     : SPI Interface
9 // Last revision   : 23.09.2023 14:13:07
10 ////////////////////////////////////////////////////////////////////
11 module spi #(parameter CLK_DIV = 2)(
12     input clk,
13     input rst,
14     input miso,
15     output mosi,
16     output sck,
17     input start,
18     input[7:0] data_in,
19     output[7:0] data_out,
20     output busy,
21     output new_data
22 );
23
24 localparam STATE_SIZE = 2;
25 localparam IDLE = 2'd0,
26 WAIT_HALF = 2'd1,
27 TRANSFER = 2'd2;
28
29 reg [STATE_SIZE-1:0] state_d, state_q;
30
31 reg [7:0] data_d, data_q;
32 reg [CLK_DIV-1:0] sck_d, sck_q;
33 reg mosi_d, mosi_q;
34 reg [2:0] ctr_d, ctr_q;
35 reg new_data_d, new_data_q;
36 reg [7:0] data_out_d, data_out_q;
37
38 assign mosi = mosi_q;
39 assign sck = (~sck_q[CLK_DIV-1]) & (state_q == TRANSFER);
40 assign busy = state_q != IDLE;
41 assign data_out = data_out_q;
42 assign new_data = new_data_q;
43
44 always @(*) begin
45     sck_d = sck_q;
46     data_d = data_q;
47     mosi_d = mosi_q;
```

# Маршрут разработки проекта



Система цифрового моделирования предоставляет разработчикам средства для разработки высокоуровневого описания ПЛИС и их логической верификации. Частично поддержаны синтез и последующие этапы разработки.

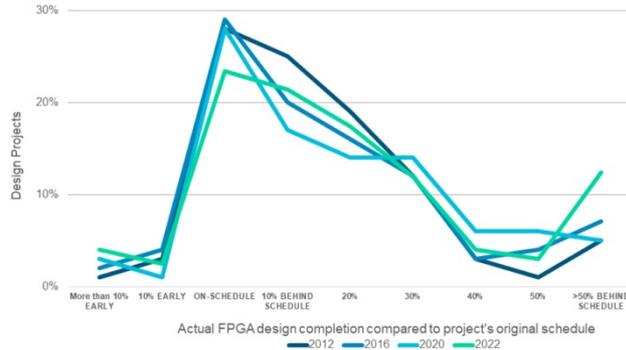
# Логическая верификация

70%

Projects behind schedule

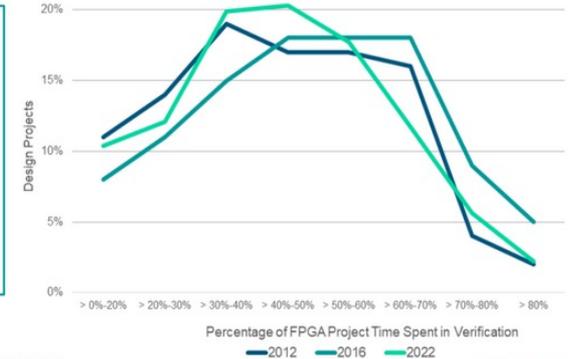
12%

Projects behind schedule more than 50%



40%-50%

Median project time spent in verification



Source: Wilson Research Group and Siemens EDA, 2022 Functional Verification Study  
Page 8 Unrestricted | © Siemens 2022 | Functional Verification Study

Source: Wilson Research Group and Siemens EDA, 2022 Functional Verification Study  
Unrestricted | © Siemens 2022 | Functional Verification Study

SIEMENS

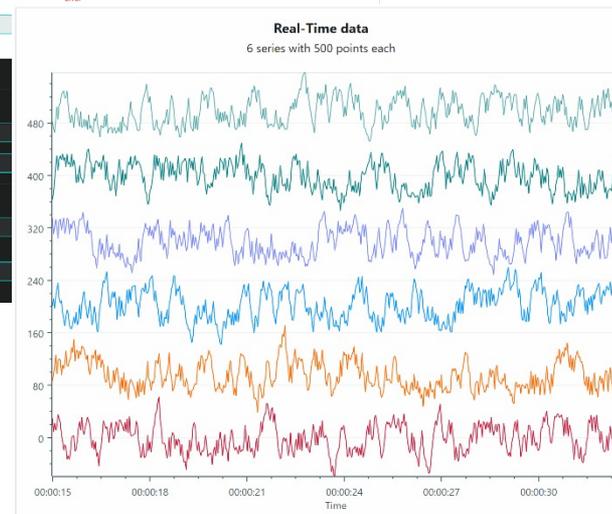
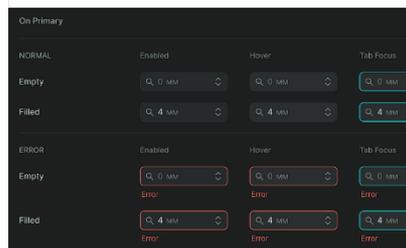
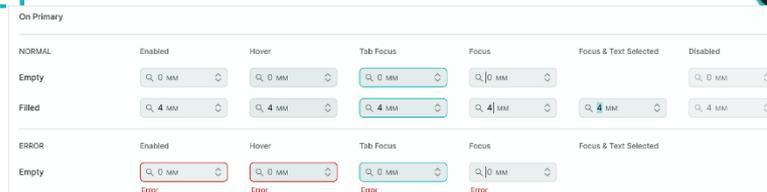
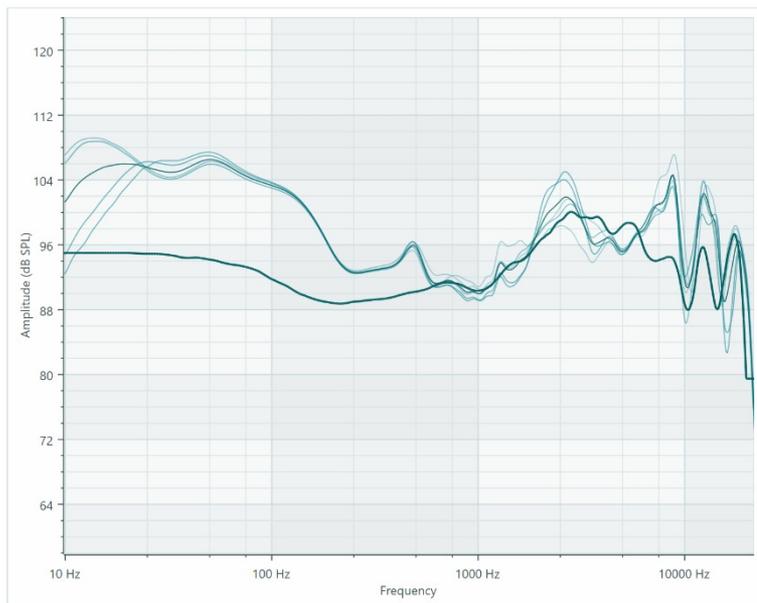
SIEMENS

Задача верификации – создание тестового окружения для поиска и исправления ошибок в дизайне проекта. Под логической верификацией проекта понимают:

1. Проверка соответствия работы дизайна по ТЗ(функциональная верификация),
2. Проверка эквивалентности свойств дизайна с эталонными значениями (формальная верификация),
3. Статический анализ кода (проверка соответствия исходного кода правилам/стандартам),

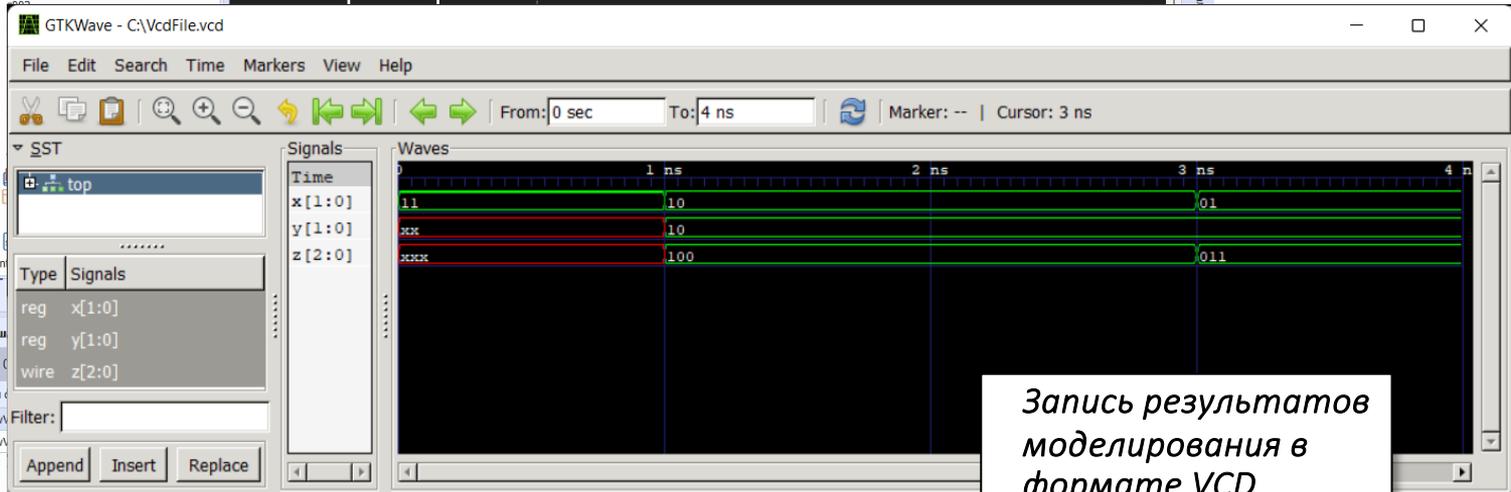
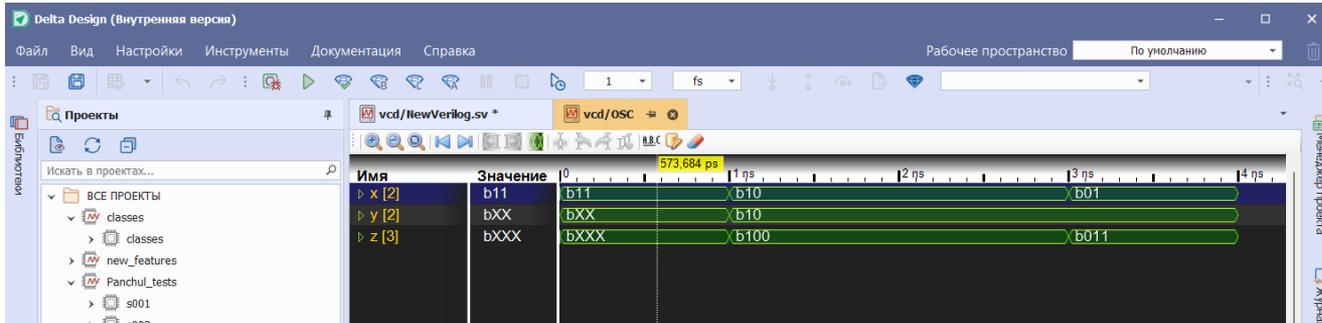
- Предоставление среды разработки проекта цифрового моделирования:
  - Описание проекта в схемотехническом виде;
  - Разработка проекта с помощью HDL-описания.
- Импорт/экспорт проектов цифрового моделирования;
- Моделирование HDL-проектов:
  - VHDL-2008;
  - SystemVerilog-2005 (частично – SystemVerilog-2023).
- Синтез проектов с помощью Yosys;
- Синтез проектов собственными инструментам;
- Подготовка файлов конфигурации для ПЛИС М3 Миландр;
- Программирование ПЛИС М3.

# Библиотека EremexControls.NET



Набор контроллов, поддержка осуществляется ОС Windows, Ubuntu, AstraLinux, RedOS, а также их модификации под Эльбрус

<https://habr.com/ru/articles/805081/>  
<https://habr.com/ru/articles/789322/>



Проекты

Искать в проектах...

- Моделирование
- Цифровое моделирование
  - alu74181
  - Butterworth Filter (AMS)
    - Butterworth Filter (AMS)
      - Технология
      - Осциллографы
        - OSC
      - Схема и результаты моделирования
      - README
        - cap.vams
        - discipline.vams
        - ind.vams
        - res.vams
        - signal.vams
        - top.vams**
      - FIR
      - schematic
      - schoolMIPS
      - ToroR
      - Коммутатор управления ШД-8 (CPLD)
      - Плата управления

Библиотеки

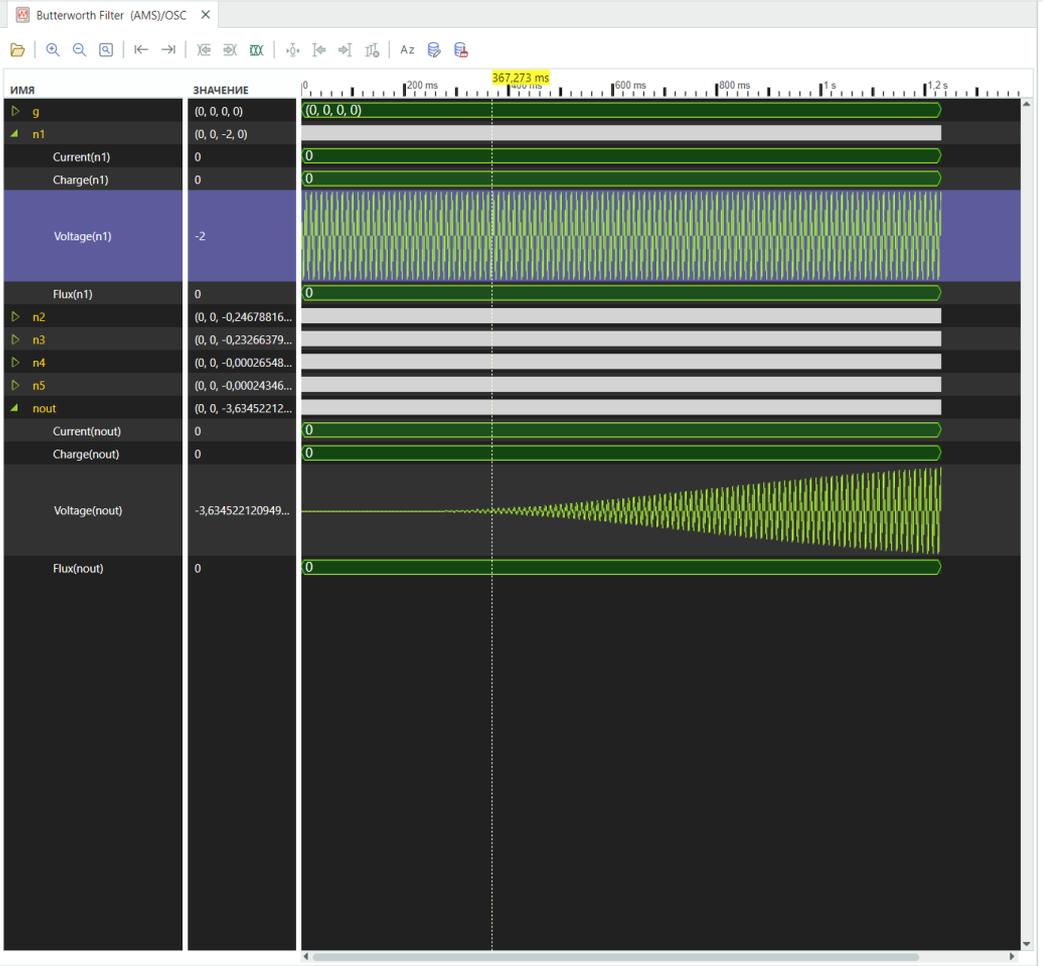
Искать в библиотеках...

- Демо библиотека
- infrastructure
- Simtera

Журналы | Список ошибок

```

1 ///////////////////////////////////////////////////////////////////
2 ///////////////////////////////////////////////////////////////////
3 //top.vams/////////////////////////////////////////////////////////////////
4 ///////////////////////////////////////////////////////////////////
5 `timescale 1ms/100us
6 module top();
7 electrical n1, n2, n3, n4, n5, nout, g;
8 ground g;
9
10 signal1 ..... V1 (n1, g);
11 //res --- #(.R(1)) ..... R1 (n1, nout);
12 //ind --- #(.L(4.09852818u)) --- L1 (nout, g);
13 //cap --- #(.C(0.618033978)) --- C1 (nout, g);
14
15 res --- #(.R(1)) ..... R1 (n1, n2);
16 ind --- #(.L(4.09852818u)) --- L1 (n2, g);
17 cap --- #(.C(0.618033978)) --- C1 (n2, g);
18 ind --- #(.L(1.618033969)) --- L2 (n2, n3);
19 cap --- #(.C(1.56549845u)) --- C2 (n3, n4);
20 ind --- #(.L(1.266514838u)) --- L3 (n4, g);
21 cap --- #(.C(2)) --- C3 (n4, g);
22 ind --- #(.L(1.61803398875)) L4 (n4, n5);
23 cap --- #(.C(1.56549845u)) --- C4 (n5, nout);
24 ind --- #(.L(4.09852818u)) --- L5 (nout, g);
25 cap --- #(.C(0.61803398875)) C5 (nout, g);
26 res --- #(.R(1)) ..... R2 (nout, g);
27
28 initial begin
29   $display($realtime, ":", V(n1), ",", V(nout));
30   for (int i = 0; i < 2459; ++i)
31     #0.5;
32   for (int i = 0; i < 10; ++i) begin
33     #0.5;
34     $display($realtime, ":", V(n1), ",", V(n2));
35   end;
36
37 end
38
39 endmodule
    
```



Перевод абстрактной поведенческой модели на уровень регистровых передач.

Шаги:

1. Лексический процессинг,
2. Анализ потока данных,
3. Библиотечная обработка,
4. Привязка функционального блока,
5. Обработка вывода,
6. Перегруппировка входных данных

## Логический синтез:

1. Получение списка соединений логических вентилей,
2. Выделение комбинаторной и последовательностной (секвенциальной) логики,
3. Применение заданных ограничений дизайна проекта,
4. Оптимизация комбинаторной логики,
5. Выделение и оптимизация КА (FSM), содержащей и комбинационную, и последовательностную логику

## Технологическое отображение:

1. Отображение дизайна проекта в элементы технологической библиотеки,
2. Обработка сторонних библиотек,
3. Анализ ячеек с учетом весов (основных характеристик: площадь, питание, скорость),
4. Перегруппировка нетлиста логического синтеза с использованием ячеек библиотек

\*Логический синтез и технологическое отображение минимально, но частично зависимы друг от друга

Название	Левая сторона	Правая сторона	Глубина	Условие
«Нейтралитет»	$a \wedge T$	$a$	1	
Ограничение	$a \wedge \perp$	$\perp$	1	
Идемпотентность	$a \wedge b$	$a$	1	$a = b$
Противоречие	$a \wedge b$	$\perp$	1	$a \neq b$
Противоречие	$(a \wedge b) \wedge c$	$\perp$	2	$(a \neq c) \vee (b \neq c)$
Противоречие	$(a \wedge b) \wedge (c \wedge d)$	$\perp$	2	$(a \neq c) \vee (a \neq d) \vee (b \neq c) \vee (b \neq d)$
Поглощение	$\neg(a \wedge b) \wedge c$	$c$	2	$(a \neq c) \vee (b \neq c)$
Поглощение	$\neg(a \wedge b) \wedge (c \wedge d)$	$c \wedge d$	2	$(a \neq c) \vee (a \neq d) \vee (b \neq c) \vee (b \neq d)$
Идемпотентность	$(a \wedge b) \wedge c$	$a \wedge b$	2	$(a = c) \vee (b = c)$
Разрешение	$\neg(a \wedge b) \wedge \neg(c \wedge d)$	$\neg a$	2	$(a = d) \wedge (b \neq c)$
Поглощение	$\neg(a \wedge b) \wedge c$	$\neg a \wedge b$	3	$b = c$
Поглощение	$\neg(a \wedge b) \wedge (c \wedge d)$	$\neg a \wedge (c \wedge d)$	3	$b = c$
Идемпотентность	$(a \wedge b) \wedge (c \wedge d)$	$(a \wedge b) \wedge d$	4	$(a = c) \vee (b = c)$
Идемпотентность	$(a \wedge b) \wedge (c \wedge d)$	$a \wedge (c \wedge d)$	4	$(b = c) \vee (b = d)$
Идемпотентность	$(a \wedge b) \wedge (c \wedge d)$	$(a \wedge b) \wedge c$	4	$(a = d) \vee (b = d)$
Идемпотентность	$(a \wedge b) \wedge (c \wedge d)$	$b \wedge (c \wedge d)$	4	$(a = c) \vee (a = d)$

На текущий момент реализованы функции ограничения, идемпотентности, противоречия, поглощения для O1 и O2

Идемпотентность – свойство операции давать тот же результат каждый раз, что и при первом применении

Реализация на глубину 2 показывает достаточную оптимизацию на некоторых проектах. В последующем – будет реализована оптимизация на большую глубину. На глубине 2 – новые узлы не создаются

	k	Число AIG узлов					Сокращение относительно O1			
		O1	O2	O3	O4	Og	O2	O3	O4	Og
eijk.S298.S	58	257579	<b>257351</b>	<b>257351</b>	<b>257351</b>	260744	0.1%	0.1%	0.1%	-1.2%
eijk.S953.S	7	10311	<b>10236</b>	<b>10236</b>	<b>10236</b>	11791	0.7%	0.7%	0.7%	-14.4%
eijk.S820.S	11	18266	18111	<b>18071</b>	18091	19843	0.8%	1.1%	1.0%	-8.6%
eijk.S510.S	10	14519	14375	14375	<b>14365</b>	16010	1.0%	1.0%	1.1%	-10.3%
eijk.S832.S	11	19434	<b>19194</b>	<b>19194</b>	19215	21040	1.2%	1.2%	1.1%	-8.3%
cmu.periodic.N	96	733095	733095	<b>719724</b>	<b>719724</b>	721452	0.0%	1.8%	1.8%	1.6%
nusmv.guid7.C	27	155260	154203	<b>152069</b>	<b>152069</b>	167223	0.7%	2.1%	2.1%	-7.7%
ken.oop1.C	29	65855	64605	<b>63674</b>	<b>63674</b>	65444	1.9%	3.3%	3.3%	0.6%
nusmv.guid1.C	10	28889	28520	<b>27721</b>	<b>27721</b>	32313	1.3%	4.0%	4.0%	-11.9%
nusmv.tcas2.B	6	25999	24657	24225	<b>24198</b>	26254	5.2%	6.8%	6.9%	-1.0%
nusmv.tcas3.B	5	20178	19023	18644	<b>18618</b>	20409	5.7%	7.6%	7.7%	-1.1%
vis.prodc24.E	37	297791	289394	<b>257570</b>	<b>257570</b>	270317	2.8%	13.5%	13.5%	9.2%
vis.prodc12.E	29	204807	198235	<b>173352</b>	<b>173352</b>	183219	3.2%	15.4%	15.4%	10.5%
vis.prodc17.E	27	183883	177779	<b>154670</b>	<b>154670</b>	163807	3.3%	15.9%	15.9%	10.9%
vis.prodc15.E	23	144774	139602	<b>120066</b>	<b>120066</b>	127763	3.6%	17.1%	17.1%	11.8%
vis.prodc19.E	22	135975	131023	<b>112273</b>	<b>112273</b>	119722	3.6%	17.4%	17.4%	12.0%
texas.par2.E	2	1009	992	813	813	872	1.7%	19.4%	19.4%	13.6%
vis.prodc14.E	16	86137	82589	<b>69211</b>	<b>69211</b>	74441	4.1%	19.7%	19.7%	13.6%
vis.prodc18.E	13	64185	61385	<b>50755</b>	<b>50755</b>	54901	4.4%	20.9%	20.9%	14.5%
vis.prodc13.E	8	33849	32161	<b>25788</b>	<b>25788</b>	28195	5.0%	23.8%	23.8%	16.7%
vis.prodc16.E	5	18217	17229	<b>13510</b>	<b>13510</b>	14819	5.4%	25.8%	25.8%	18.7%

## Улучшение инструментов симуляции



- Добавлена поддержка структур,
- Работа именных событий,
- String data type,
- case\_inside\_items,
- unique\_case, unique0\_case, priority\_case,
- Ассоциативные массивы и методы работы с ними,
- На порядок выросла скорость компиляции и симуляции,
- Улучшения и доработки по >80 пунктам
- **Поддержка Verilog-AMS**

- Определение шаблонов синтеза памяти по исходному дизайну,
- Генерация выходного нетлиста в формате Verilog,
- Формирование и наполнение логических конусов,
- Определение конечных автоматов в исходном дизайне,
- Улучшения поиска и генерации последовательной логики,
- Стабилизация работы синтеза и увеличение количества поддерживаемых конструкций языка Verilog,
- Улучшения и доработки по >100 пунктам
- Увеличение количества вентиляей до 30 тыс.

LEC (проверка логической эквивалентности)\*

STA (статический временной анализ)\*

\* - альфа-тестирование будет объявлено позже

Собственная платформа позволяет:

1. Проводить синтез под любые ПЛИС при наличии технологической библиотеки.
2. Присутствуют собственные инструменты по трассировке/ автотрассировке (в наст. время применяются в ПП), графическое ядро, API, база данных, менеджер библиотек и пр.
3. На текущий момент поддерживает ПЛИС МЗ (Миландр)

Проекты

Искать в проектах...

- ВСЕ ПРОЕКТЫ
- Примеры
  - CAM
  - TopoR
  - Моделирование
  - Цифровое моделирование
    - 1
    - 12
      - 12(1)
        - Осциллографы и Списки наблюдений
          - Tree
            - NewVerlog\_1.txt
            - NewVerlog\_2.txt
            - NewVerlog\_2\_1.txt
            - NewVerlog\_3\_1.txt
            - NewVerlog\_3\_2.txt
            - NewVerlog\_4\_1.txt
            - NewVerlog\_5\_1.txt
            - NewVerlog\_6\_1.txt
            - NewVerlog\_7\_1.txt
            - NewVerlog.v
            - helloworld\_synth.edf
            - helloworld\_synth.v
            - helloworld\_synth\_1.edf
            - helloworld\_synth\_1.v
            - helloworld\_synth\_2.edf
            - helloworld\_synth\_2.v
            - helloworld\_synth\_3.edf
            - helloworld\_synth\_3.v
            - YosysDotFile\_helloworld.dot
            - YosysDotFile\_helloworld\_1.dot
            - YosysDotFile\_helloworld\_2.dot
- FIR\_UART
  - Схема
    - FIR
      - Осциллографы и Списки наблюдений
        - FIRBit.vhd
        - FIRBit\_TB.vhd
      - UART
- X
- milandr\_test
- model
  - model(1)
    - Осциллографы и Списки наблюдений
      - OSC
      - Tree
        - NewVerlog.v
  - SPI Master/Slave
  - at88sc01
  - at88sc01reader
  - FPDA\_XCLOS2 devKit
  - Компьютерное управление ШД-8 (CPLD)
  - Плата управления
  - Сканер CAN-bus
  - Цифровой преобразователь температуры

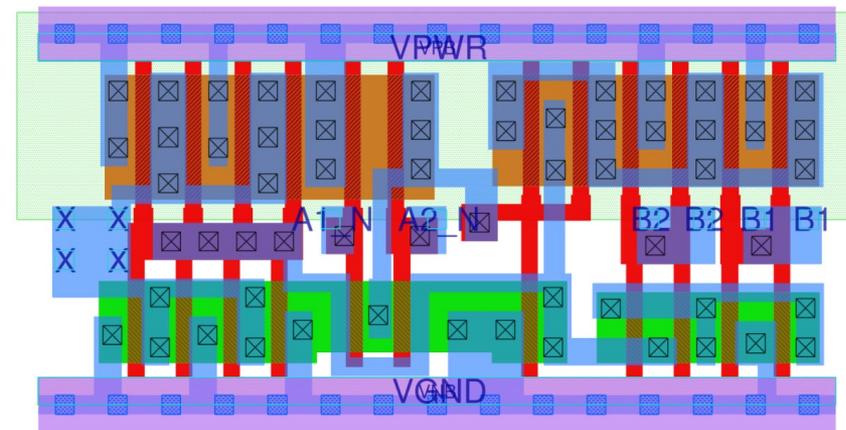
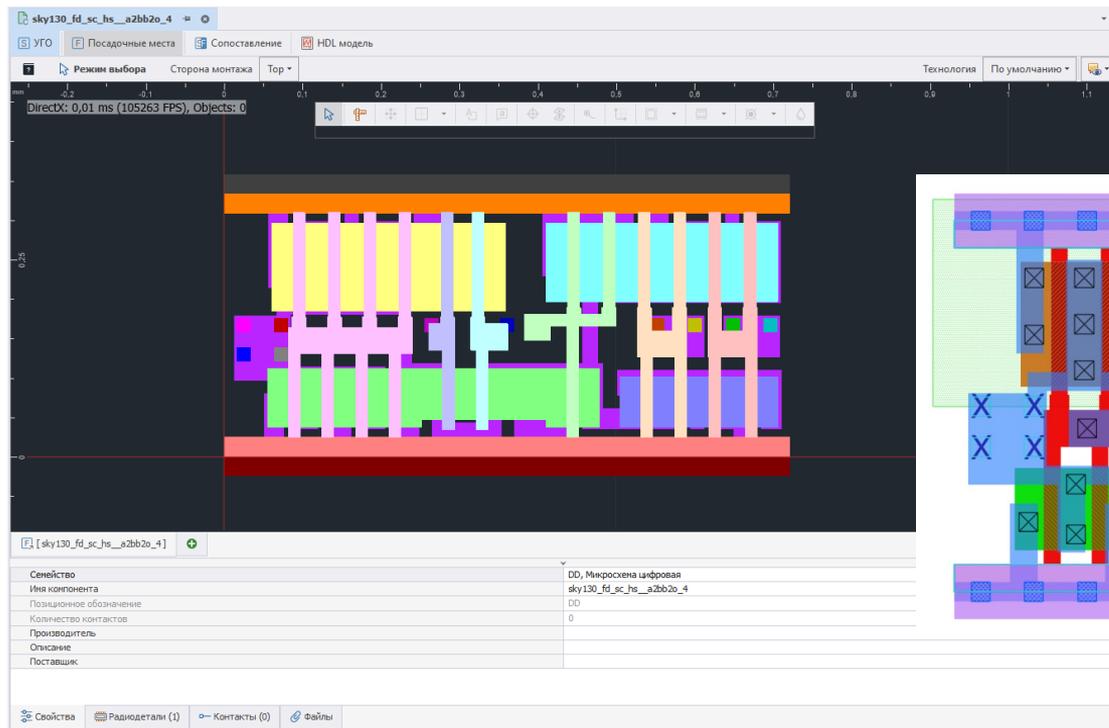


Milandr PLISM3\_DEVICE\_8307.144-AH3

Имя порта	Имя вывода	Ток вывода	Логический уровень	Триггер Шаттла	Поддержка вывода	Номинал резистора
clk_1	U1 (B1)	8.0 mA	U3V3COMPLT	off	off	50 KOhm
led0	U2 (A1)	8.0 mA	U3V3COMPLT	off	off	50 KOhm
led1	U29 (M1)	8.0 mA	U3V3COMPLT	off	off	50 KOhm

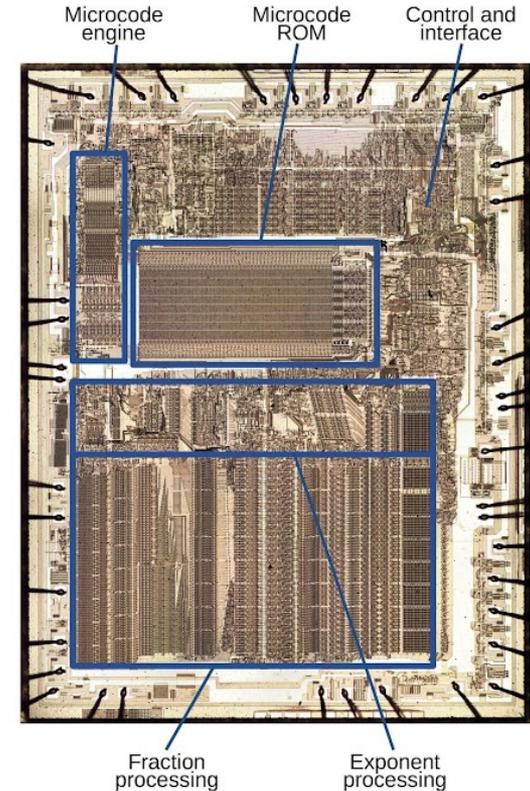
Принять ограничения Очистить таблицу

# Импорт Process Design Kit (PDK)



# Планы

- Расширение синтезируемого подмножества Verilog
- Подготовка сравнительного анализа на бенчмарках SchoolMips - <https://github.com/MIPSFpga/schoolMIPS>  
Intel 8051  
Intel 8086 - <https://github.com/jamieiles/80x86>  
с увеличением количества ячеек в проекте до 40 тыс.
- Расширение поддержки атрибутов при исполнении синтеза
- Увеличение глубины оптимизации с перезаписью узлов на глубину  $> 2$
- Реализация оптимизация на первичном «поведенческом» уровне





## Simtera Support

- Публикация новых патчей;
- Пользовательская поддержка



Спасибо за внимание!

[www.eremex.ru](http://www.eremex.ru)



Генеральный партнёр конференции FPGA-Systems  
2024.1



Первая современная отечественная САПР,  
реализующая сквозной цикл проектирования печатных плат

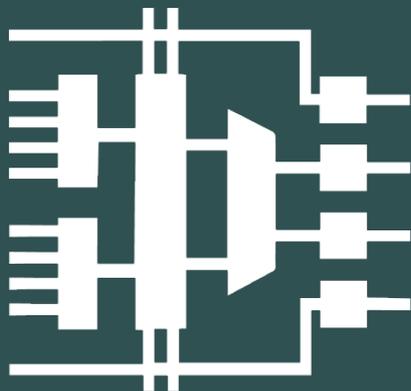


[www.aremex.ru](http://www.aremex.ru)

# Где найти FPGA / RTL / Verification комьюнити?

[FPGA-Systems.ru](http://FPGA-Systems.ru)

Сайт комьюнити



[FPGA-Systems Magazine \(FSM\)](#)

Первый журнал о программируемой логике

[@fpgasystems](#)

Телеграм чат

[admin@fpga-systems.ru](mailto:admin@fpga-systems.ru)

Электронная почта

[Youtube.com/c/fpgasystems](https://Youtube.com/c/fpgasystems)

Youtube канал