

# Статическая проверка конфигурации системы логических анализаторов микропроцессоров семейства «Эльбрус»

Выполнил студент группы М01-903а Самирханов Д.И.

Московский физико-технический институт  
(государственный университет)  
Физтех-школа радиотехники и компьютерных технологий  
Кафедра информатики и вычислительной техники

2019

# Проблема

- ▶ В микропроцессорах «Эльбрус» используются собственные стандартные модули логических анализаторов (ЛА), позволяющие осуществлять сбор и анализ временных диаграмм заведенных на них сигналов;
- ▶ ответственный инженер должен вручную подключить наблюдаемые сигналы и описать их;
- ▶ как правило, ошибки подключения обнаруживаются при отладке устройства;
- ▶ требуется верификация подключения на этапе разработки: необходимо сопоставить RTL и описание подключенных сигналов в заданном формате;
- ▶ в связи с тем, что в документации отсутствуют полные иерархические пути сигналов, необходимо применить статический анализ verilog-описания.

## Цель работы

Разработка инструмента статической проверки конфигурации система ЛА микропроцессоров семейства «Эльбрус».

Задачи:

- ▶ проверка целостности цепи ЛА;
- ▶ анализ подключенных сигналов на соответствие документации:
  - ▶ экстракция из RTL-описания списка подключенных к ЛА сигналов;
  - ▶ сравнение имен и параметров сигналов с задокументированными.

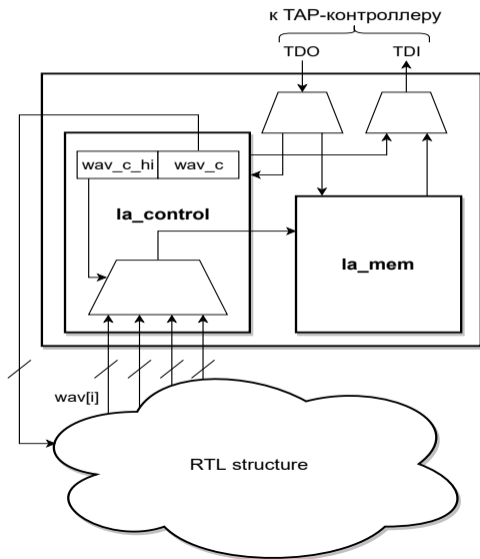
Требование: для статического анализа RTL-описания использовать VPI (Verilog Procedural Interface).

# VPI

VPI — стандартный интерфейс симулятора RTL-описания, который позволяет сторонним приложениям получить доступ к внутренней структуре моделируемого устройства.

- ▶ Не требуется самостоятельно производить лексический разбор verilog-файлов RTL-описания;
- ▶ каждому verilog-объекту ставится в соответствие дескриптор;
- ▶ интерфейс позволяет получать доступ к свойствам verilog-объектов при помощи их дескрипторов;
- ▶ связи между исследуемыми объектами разделяются на 2 вида:
  1. физические соединения (например между шиной и портом);
  2. логические связи (доступ к свойствам объекта, таким как разрядность или имя);

# Структура модуля ЛА



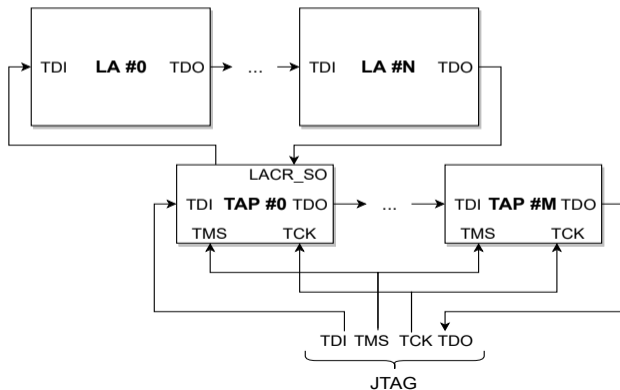
- ▶ Подмодуль **la\_mem**: хранение данных;
- ▶ подмодуль **la\_control**: управление;
- ▶ шины **wav[i]**: подключение наблюдаемых сигналов;
- ▶ регистр **wav\_c\_hi**: выбор активной шины данных;
- ▶ шина **wav\_c**: выбор набора сигналов, подключаемого к активной шине данных.

# Проверка целостности цепи ЛА

## Подключение модулей ЛА к TAP-контроллерам

- ▶ Последовательность ЛА формирует сдвиговый регистр LACR TAP-контроллеров;
- ▶ LACR подключен ко входу LACR\_SO TAP-контроллера.

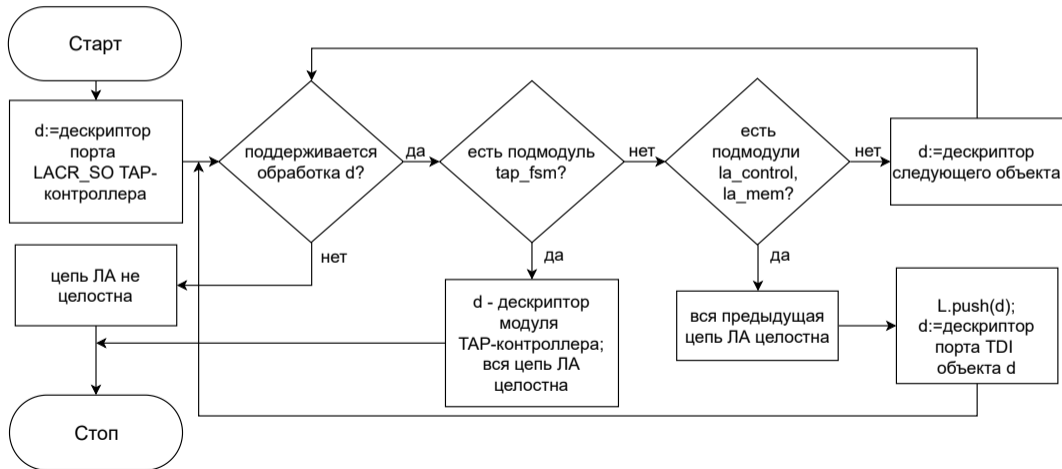
Для заданного TAP-контроллера необходимо проверить целостность цепи ЛА, составить список анализаторов и сравнить его с документацией.



# Проверка целостности цепи ЛА

## Реализованный алгоритм

d — дескриптор исследуемого объекта; L — список дескрипторов найденных ЛА.

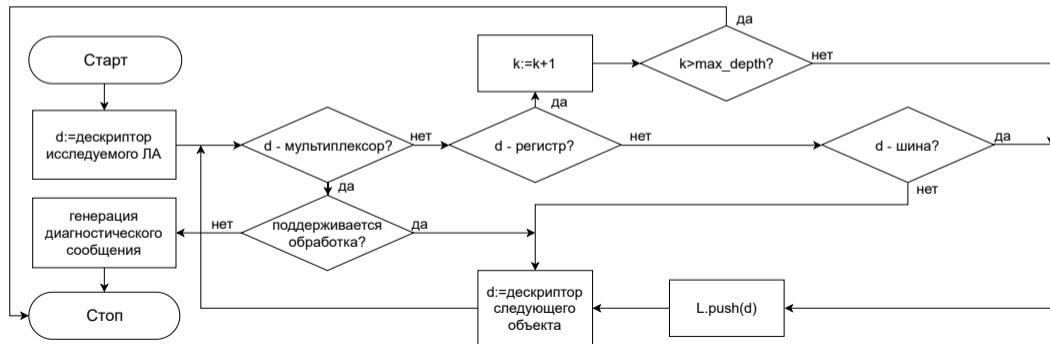


# Экстракция из RTL-описания списка подключенных к ЛА сигналов

## Реализованный алгоритм

$d$  — дескриптор объекта;  $L$  — список дескрипторов найденных сигналов;  $k$  — счетчик глубины обхода;  $\text{max\_depth}$  — заданный параметр глубины обхода.

Для каждого бита активной шины данных осуществляем обход RTL-описания по физическим соединениям в направлении от нагрузки к источнику сигнала





# Экстракция из RTL—описания списка подключенных к ЛА сигналов

## Обработка мультиплексоров

- ▶ Способ подключения сигналов к ЛА не стандартизован;
- ▶ необходимо определить провод-источник сигнала для заданного выходного разряда;
- ▶ поддерживаемые типы мультиплексоров:

```
wav[1] <= (({68{(wav_c == 5'h00)}}&net_1)|  
          ({68{(wav_c == 5'h01)}}&net_2);
```

```
case (wav_c):  
1: wav[1] <= net\_1;  
2: wav[1] <= net\_2;  
default: wav[1] <= 68'h00;
```

```
wav[1] <= (wav_c == 5h'00)? net_1:net_2;
```

## Сравнение имен и параметров сигналов с задокументированными

Для каждого объекта из экстрагированного списка источников осуществляется сравнение с соответствующим сигналом из документации и проверка:

- ▶ вхождения имени сигнала из документации в иерархический путь дескриптора с учетом разрядности;
- ▶ суммарной разрядности шин;
- ▶ соответствия порядка сигналов и бит шин данных.

При каждом несоответствии выводится диагностическое сообщение.

## Дальнейшая работа

Предложенный алгоритм разбора мультиплексоров произвольного типа

Для верификации корректности работы логики разработан и предложен следующий алгоритм:

1. для входов мультиплексоров RTL-описания принудительно устанавливаем определенные наборы логических значений;
2. ожидаем, пока не обновятся значения на битах активных шин данных, и считываем их;
3. проверяем корректность полученного результата.

# Результаты

- ▶ Реализован инструмент статической проверки конфигурации система ЛА микропроцессоров семейства «Эльбрус»;
- ▶ реализована поддержка 3 основных типов мультиплексоров сигналов;
- ▶ предложен алгоритм разбора мультиплексоров произвольного типа;
- ▶ подтверждена целостность всех цепочек ЛА ТАР-контроллеров процессоров Эльбрус-8с2 и R-2000;
- ▶ найдены и воспроизведены ошибки в описании подключенных сигналов для микропроцессора R-2000;
- ▶ на примере изготовленного микропроцессора подтверждена применимость подхода: ошибка, найденная при отладке, была обнаружена разработанным инструментом.