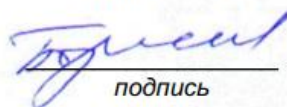


МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ВГУ»)

УТВЕРЖДАЮ

Заведующий кафедрой
физики полупроводников и микроэлектроники


подпись

(Бормонтов Е.Н.)
расшифровка подписи

31.08.2024

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

по учебной дисциплине

Б1.В.01 Проектирование цифровых устройств на Verilog

Код и наименование направления подготовки/специальности: 11.04.04

Электроника и наноэлектроника

Направленность (профиль) подготовки/специализация: _____

Интегральная электроника и наноэлектроника

Квалификация выпускника: магистр

Форма обучения: очная

Кафедра, отвечающая за реализацию дисциплины: _____

физики полупроводников и микроэлектроники

Составители рабочей программы дисциплины, в том числе фонда оценочных средств по учебной дисциплине: Богатиков Евгений Васильевич, кандидат физико-математических наук, доцент

Учебный год
освоения дисциплины: 2024-2025

Семестр(ы): 2

Освоение данной дисциплины направлено на формирование следующих-компетенций

Компетенции		Индикаторы		Планируемые результаты обучения
Код	Наименование компетенции	Код(ы)	Наименование индикатора(ов)	
ПК-3	Способен анализировать состояние научно-технической проблемы путём подбора, изучения и анализа литературных и патентных источников	ПК-3.1	Составляет описание алгоритма функционирования и циклограммы работы СнК и формулирует предложения по их реализации аппаратными или программными средствами	Знает: - архитектуру ПЛИС - принципы разработки HDL-описаний цифровых устройств; Умеет: - составлять описание алгоритма функционирования и циклограммы работы СнК; Владеет: - программными средствами реализации цифровых устройств
		ПК-3.2	Разрабатывает технические требования к созданию аналоговых и аналого-цифровых узлов СнК	Умение: - применять язык Verilog для создания синтезируемых HDL-описаний цифровых устройств по техническому заданию; - разрабатывать методику тестирования и проводить функциональную верификацию HDL-описания; - проводить конфигурирование ПЛИС и тестирование законченного цифрового устройства на базе ПЛИС.
		ПК-3.3	Проводит технико-экономический анализ и обосновывает принимаемые решения по выбору архитектуры СнК	Умеет: - выполнять технико-экономический анализ по выбору архитектуры СнК; - обосновывать принимаемые решения по выбору архитектуры СнК
ПК-7	Готов определять цели, осуществлять постановку задач проектирования электронных приборов, схем и устройств различного функционального назначения, подготавливать технические задания на выполнение проектных работ	ПК-7.1	Производит выбор языков описания аппаратуры и стилей описания цифровых блоков, а также выбор средств описания поведенческих моделей аналоговых блоков	Знает: - языки описания аппаратуры; Умеет: - использовать языки описания аппаратуры при проектировании цифровых блоков СнК
		ПК-7.2	Формулирует задачи функциональной и временной верификации цифровых блоков СнК, производит выбор методики верификации поведенческих моделей аналоговых	Знает: - методики верификации цифровых блоков; Умеет: - формулировать задачи функциональной и временной верификации цифровых блоков СнК; - производить выбор методики верификации цифровых блоков

			блоков	
ПК-8	Способен проектировать устройства, приборы и системы электронной техники с учетом заданных требований	ПК-8.1	Использует языки описания аппаратуры при проектировании цифровых и аналоговых блоков СнК	Знает: - языки описания аппаратуры; Умеет: - использовать языки описания аппаратуры при проектировании цифровых блоков СнК
		ПК-8.2	Моделирует средствами САПР функциональное описание цифровых блоков и использует его результаты для коррекции их функционального описания	Умеет: - применять язык Verilog для создания синтезируемых HDL-описаний цифровых устройств по техническому заданию; - разрабатывать методику тестирования и проводить функциональную верификацию HDL-описания; - проводить конфигурирование ПЛИС и тестирование законченного цифрового устройства на базе ПЛИС.

Перечень заданий для оценки уровня освоения дисциплины:

1) тестовые задания:

1. При помощи какого оператора языка Verilog можно изменять значение сигнала типа wire?

Варианты ответа:

- A) assign
- B) always
- C) initial
- D) case
- E) while

Правильный ответ: А.

2. Какие из операторов языка Verilog являются параллельными?

Варианты ответа:

- A) assign
- B) always
- C) case
- D) if
- E) for

Правильный ответ: А, В.

3. Какие операторы языка Verilog являются последовательными?

Варианты ответа:

- A) case
- B) for
- C) always
- D) assign
- E) initial

Правильный ответ: A, B.

4. Какая языковая конструкция Verilog не является синтезируемой?

Варианты ответа:

- A) always #5 a=~b;
- B) assign a=~b;
- C) always @(negedge b) a<=~c;
- D) assign a=(b<c)?0:1;
- E) assign a=(b>c)?1'b1:1'bz;

Правильный ответ: A.

5. Какое утверждение относительно портов модуля тестбенча является верным?

Варианты ответа:

- A) модуль тестбенча не содержит портов
- B) в модуле тестбенча не должно быть входов, но должны быть выходы
- C) в модуле тестбенча не должно быть выходов, но должны быть входы
- D) модуль тестбенча должен иметь и входы, и выходы
- E) модуль тестбенча должен иметь только двунаправленные порты

Правильный ответ: A.

6. Найдите неверное утверждение про оператор initial.

Варианты ответа:

- A) оператор initial должен располагаться перед операторами assign и always
- B) оператор initial используется при написании тестбенчей
- C) оператор initial относится к параллельным операторам
- D) оператор initial запускается на выполнение один раз
- E) внутри оператора initial можно настроить сохранение результатов и продолжительность моделирования

Правильный ответ: A.

7. Выберите, какому элементу соответствует следующий код на языке Verilog:

```
module nemo(a,b,c);  
  input a,b;  
  output reg c;  
  always @(negedge a) c<=b;  
endmodule
```

Варианты ответа:

- A) D-триггер, тактируемый по заднему фронту
- B) D-триггер, тактируемый по переднему фронту
- C) мультиплексор
- D) сдвиговый регистр
- E) двунаправленный буфер

Правильный ответ: A.

8. Выберите, какому элементу соответствует следующий код на языке Verilog:

```
module nemo (a, b, c);
  input a, b;
  output c;
  assign c = a ? b : 1'bz;
endmodule
```

Варианты ответа:

- A) буфер с z-состоянием
- B) компаратор
- C) RS-триггер
- D) мультиплексор
- E) приоритетный шифратор

Правильный ответ: A.

9. Выберите, какому элементу соответствует следующий код на языке Verilog:

```
module nemo (a, b, c);
  input [1:0] a;
  input [3:0] b;
  output c;
  reg d;
  always @(a)
  begin
    case (a)
      2'b00 : d<=b[0];
      2'b01 : d<=b[1];
      2'b10 : d<=b[2];
      2'b11 : d<=b[3];
    endcase
  end
  assign c = d;
endmodule
```

Варианты ответа:

- A) мультиплексор
- B) триггер
- C) сдвиговый регистр
- D) компаратор
- E) тождественность

Правильный ответ: A.

10. Выберите, какому элементу соответствует следующий код на языке Verilog:

```
module nemo (a, b, c);
  input a, b;
  output [7:0] c;
  reg [7:0] d;
  always @ (posedge a) d<={d[6:0],b};
  assign c=d;
endmodule
```

Варианты ответа:

- A) сдвиговый регистр
- B) JK-триггер
- C) двунаправленный порт
- D) мультиплексор
- E) шифратор

Правильный ответ: А.

11. Какие из нижеперечисленных описаний портов не могут быть правильными или достаточно полными (сигналы a, b имеют тип wire)?

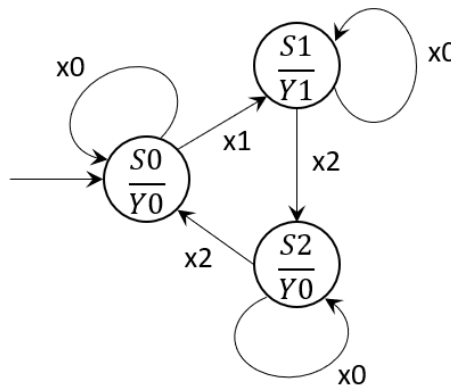
Варианты ответа:

- A) module nemo(input a, output b);
- B) module nemo(a, b);
input a;
output b;
- C) module nemo(a, b);
wire a;
wire b;
- D) module nemo(input wire a, output wire b);
- E) module nemo(a, b);

Правильный ответ: С, Е.

3) расчетные задачи, ситуационные, практико-ориентированные задачи / мини-кейсы:

12. Составьте совмещенную таблицу переходов и выходов конечного автомата, соответствующую следующему графу конечного автомата:



Решение:

Заголовочный столбец таблицы должен содержать все входные состояния автомата, а заголовочная строка – все внутренние состояния автомата. На пересечении строк и столбцов должны находиться состояния автомата, в которые он перейдет из текущего состояния (указанного в соответствующей ячейке заголовочной строки) под действием текущего входного состояния (указанного в соответствующей ячейке заголовочного столбца). Вместе с внутренними состояниями автомата указываются соответствующие выходные состояния. Для тех переходов, которым не соответствуют ребра графов, указываются прочерки.

Ответ:

X\S	S0	S1	S2
X0	S0/Y0	S1/Y1	S2/Y0
X1	S1/Y1	-	-

X2	-	S2/Y0	S0/Y0
----	---	-------	-------

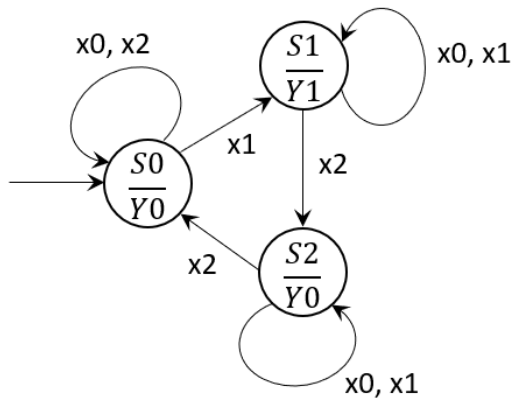
13. Изобразите граф конечного автомата, соответствующий следующей таблице переходов и выходов конечного автомата:

X\S	S0	S1	S2
X0	S0/Y0	S1/Y1	S2/Y0
X1	S1/Y1	S1/Y1	S2/Y0
X2	S0/Y0	S2/Y0	S0/Y0

Решение:

Заголовочный столбец таблицы содержит все входные состояния автомата, а заголовочная строка – все внутренние состояния автомата. На пересечении строк и столбцов находятся состояния автомата, в которые он перейдет из текущего состояния (указанного в соответствующей ячейке заголовочной строки) под действием текущего входного состояния (указанного в соответствующей ячейке заголовочного столбца). Вместе с внутренними состояниями автомата указываются соответствующие выходные состояния. Для тех переходов, которым не соответствуют ребра графов, указываются прочерки.

Ответ:



14. Составьте совмещенную таблицу переходов и выходов конечного автомата, соответствующую следующему описанию на языке Verilog:

```

module FSM (output reg Y0, Y1, input X0, X1, X2, X3, X4, clk, rst_n);
  reg [1:0] state, next;
  always @(posedge clk or negedge rst_n)
    if (!rst_n) state <= 2'b00;
    else state <= next;
  always @*
  begin
    next = 2'bx;
    case (state)
      2'b00: begin
        if (X0) next = 2'b01;
        if (X1) next = 2'b00;
        Y0 = 1'b1;
        Y1 = 1'b0;
      end
      2'b01: begin
        if (X3) next = 2'b10;
        if (X2) next = 2'b01;
        if (X4) next = 2'b11;
        Y0 = 1'b0;
      end
    endcase
  end
end
  
```

```

        Y1 = 1'b1;
    end
2'b10: begin
    if (X3) next = 2'b11;
    if (X4) next = 2'b10;
    Y0 = 1'b0;
    Y1 = 1'b1;
end
2'b11: begin
    if (X1) next = 2'b00;
    if (X0) next = 2'b01;
    Y0 = 1'b1;
    Y1 = 1'b0;
end
endcase
end
endmodule

```

Решение:

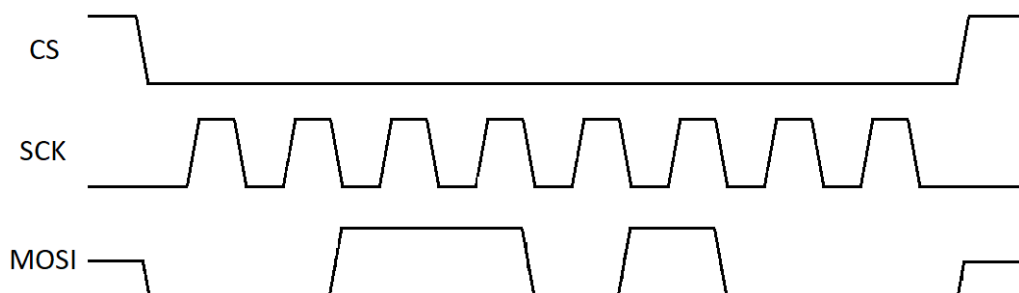
Заголовочный столбец таблицы должен содержать все входные состояния автомата, а заголовочная строка – все внутренние состояния автомата. На пересечении строк и столбцов должны находиться состояния автомата, в которые он перейдет из текущего состояния (указанного в соответствующей ячейке заголовочной строки) под действием текущего входного состояния (указанного в соответствующей ячейке заголовочного столбца). Вместе с внутренними состояниями автомата указываются соответствующие выходные состояния. Для тех переходов, которым не соответствуют ребра графов, указываются прочерки.

Количество внутренних состояний автомата определяется оператором case. В данном случае это 4 состояния. Количество входных и выходных состояний определяется строкой-заголовком модуля.

Ответ:

X\S	S0	S1	S2	S3
X0	S1/Y1	-	-	S1/Y1
X1	S0/Y0	-	-	S0/Y0
X2	-	S1/Y1	-	-
X3	-	S2/Y1	S3/Y0	-
X4	-	S3/Y0	S2/Y1	-

15. Передача данных по интерфейсу SPI в режиме Mode0 от ведущего к ведомому имеет следующую спецификацию: первый принимаемый бит игнорируется, второй бит означает порядок передаваемых данных (0 - первым передается старший бит числа, 1 - первым передается младший бит числа), следующие биты - передаваемое число. Передаче какого числа соответствует приведенная временная диаграмма? (ответ указать в десятичном формате)

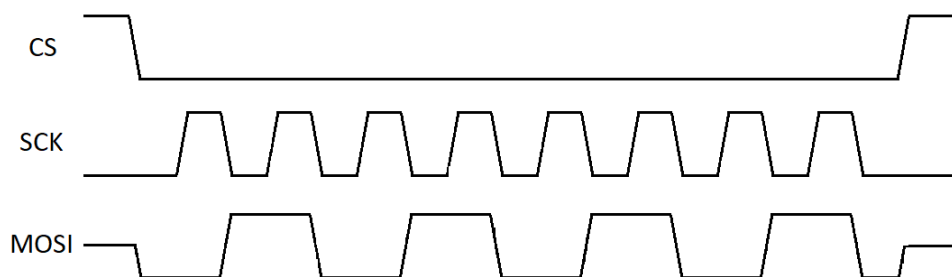


Решение:

При передаче данных по интерфейсу SPI в режиме Mode0 от ведущего к ведомому необходимо считывать данные сигнала MOSI по переднему фронту сигнала SCK. На приведенной диаграмме передается следующая последовательность бит: 0,0,1,1,0,1,0,0 (слева – первый передаваемый бит). Первый передаваемый бит согласно условию игнорируется. Равенство нулю второго передаваемого бита означает, что в последующих передаваемых битах сначала идет старший бит передаваемого числа. Таким образом, в двоичном виде было передано число 110100_2 , что соответствует числу 52_{10} в десятичной системе счисления.

Ответ: 52.

16. Передача данных по интерфейсу SPI в режиме Mode0 от ведущего к ведомому имеет следующую спецификацию: первый принимаемый бит игнорируется, второй бит означает порядок передаваемых данных (0 - первым передается старший бит числа, 1 - первым передается младший бит числа), следующие биты - передаваемое число. Передаче какого числа соответствует приведенная временная диаграмма? (ответ записать в десятичном)



Решение:

При передаче данных по интерфейсу SPI в режиме Mode0 от ведущего к ведомому необходимо считывать данные сигнала MOSI по переднему фронту сигнала SCK. На приведенной диаграмме передается следующая последовательность бит: 0,1,0,1,0,1,0,1 (слева – первый передаваемый бит). Первый передаваемый бит согласно условию игнорируется. Равенство единице второго передаваемого бита означает, что в последующих передаваемых битах сначала идет младший бит передаваемого числа. Таким образом, в двоичном виде было передано число 101010_2 , что соответствует числу 42_{10} в десятичной системе счисления.

Ответ: 42.

17. Какая временная диаграмма соответствует следующему коду на языке Verilog?

```
`timescale 1ns/1ns
module nemo();
  reg a=0;
  reg b=0;
  initial
  begin
    #1 a<=1;
    #1 a<=0;
    #1 a<=1;
    #1 a<=0;
    #1 a<=1;
    #1 $stop;
  end
```

```

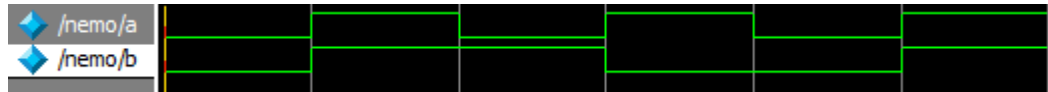
always @(posedge a) b<=!b;
endmodule

```

Решение:

В приведенной Verilog-модели сигнал «а» изменяется в операторе инициализации initial на каждом шаге моделирования, который составляет 1 нс. Изменения производятся 5 раз, после чего моделирование останавливается системной функцией \$stop. Начальное значение сигнала «а», задаваемое при объявлении сигнала, равно 0. Сигнал «b» изменяется оператором always по переднему фронту сигнала «а» на инвертированное значение сигнала «а». Начальное значение сигнала «b» равно 0.

Ответ:



18. Какая из временных диаграмм соответствует приведенному коду на Verilog?

```

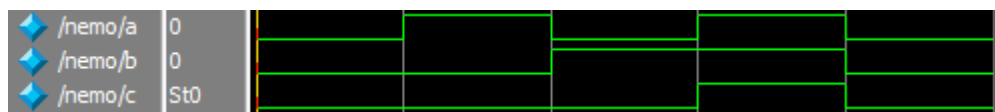
`timescale 1ns/1ps
module nemo();
  reg a=0;
  reg b=0;
  wire c;
  initial
  begin
    #5 $stop;
  end
  always #1 a <= !a;
  always #2 b <= !b;
  assign c = a & b;
endmodule

```

Решение:

В приведенной Verilog-модели сигнал «а» инвертируется в операторе процесса always через 1 нс, а сигнал «b» - через 2 нс. Начальные значения этих сигналов равны 0. Сигнал «с» изменяется оператором непрерывного присваивания assign и представляет собой результат логической операции «И» над сигналами «а» и «b».

Ответ:



Критерии и шкалы оценивания:

Для оценивания выполнения заданий используется балльная шкала:

1) тестовые задания:

- 1 балл – указан верный ответ;
- 0 баллов – указан неверный ответ, в том числе частично.

3) расчетные задачи, ситуационные, практико-ориентированные задачи / мини-кейсы:

- 5 баллов – задача решена верно (получен правильный ответ, обоснован (аргументирован) ход решения);

- 2 балла – решение задачи содержит незначительные ошибки, но приведен правильный ход рассуждений, или получен верный ответ, но отсутствует обоснование хода ее решения, или задача решена не полностью, но получены промежуточные результаты, отражающие правильность хода решения задачи, или, в случае если задание состоит из решения нескольких подзадач, 50% которых решены верно;
- 0 баллов – задача не решена или решение неверно (ход решения ошибочен или содержит грубые ошибки, значительно влияющие на дальнейшее изучение задачи).