

## **Домашнее задание №1 по курсу:**

### **«Введение в архитектуру вычислительных систем»**

В данном задании вы спроектируете одну из предложенных цифровых схем и проанализируете ее поведение. Моделирование будет происходить в программе с открытым исходным кодом «Logisim» под лицензией GNU GPL 2.0.

#### **Часть 1: Установка Logisim**

Инструкция для разных платформ описана [на официальном сайте](#).

#### **Примечания:**

1. Не забудьте установить Java (см. пункт 1)
2. На MacOS возможно потребуется дать [разрешение на установку из неизвестных источников](#).
3. Также на MacOS может быть проблема с ошибкой при запуске приложения. В таком случае нажмите ПКМ на Logisim.app -> Show Package Contents -> Contents -> Resources -> Java и осуществите запуск файла Logisim.jar. В таком случае должно сработать.

## Часть 2: Интерфейс программы и процесс разработки схемы

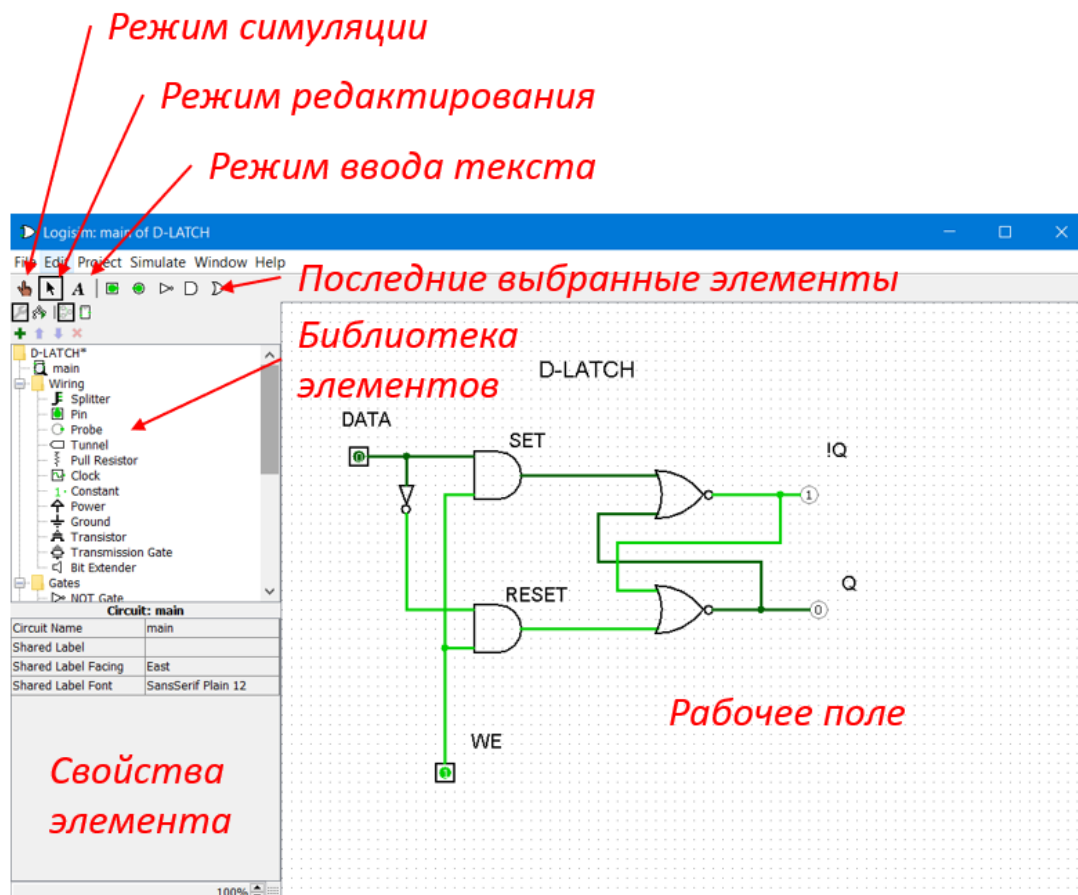


Рисунок 1 Интерфейс программы

- Построение схемы происходит из **Режима редактирования** переносом на *рабочее поле* логических элементов из меню *gates* библиотеки элементов. В **Режиме редактирования** также работает множественное выделение элементов и перемещение их по рабочему полю.
- **Поворот логического элемента** осуществляется в меню *Facing* в *Свойствах элемента* (см. рисунок 2)
- Выбор числа входов осуществляется в меню *Number of Inputs* в *Свойствах элемента*.
- Отображаемый размер выбирается в меню *Gate Size* в *Свойствах элемента*.

Selection: AND Gate	
Facing	East
Data Bits	1
Gate Size	Medium
Number Of Inputs	2
Output Value	0/1
Label	
Label Font	SansSerif Plain 12
Negate 1 (Top)	No
Negate 2 (Bottom)	No

Рисунок 2 Меню Свойство элемента

- Соединение элементов осуществляется **в Режиме редактирования** ЛКМ нажатием мыши на вход/выход и перетаскиванием провода (при этом мышь должна быть зажата). Для соединения 2 проводов – провод нужно «тащить» до «стыковки» с другим.

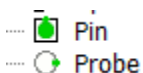


- обозначает соединение проводов,



- обозначает пересечение.

- На входы схемы** ставится элемент *Pin* (см. Иллюстрацию) из меню *Wiring* библиотеки элементов. **При этом в его свойствах в разделе «Three-state?» нужно указать «No».** *Pin* будет работать как кнопка в режиме симуляции, которая будет переключать вход с высокого сигнала на низкий и наоборот.
- На выход схемы** устанавливается элемент *Probe* (см. Иллюстрацию) из меню *Wiring* библиотеки элементов. Внутри него отображается 0 или 1 в зависимости от уровня сигнала на выходе схемы.



- Не забудьте (если нужно) повернуть *Pin* и *Probe* в меню *Facing* в *Свойствах элемента* и подключить их проводами к остальной схеме.

- После составления схемы можно перейти в **Режим симуляции** и поменять значение входов нажатием на *Pin'ы* (на входах схемы). Если после переключения состояний провода имеют **красный** цвет, значит в схеме присутствует ошибка или схема не инициализирована. Провод с низким уровнем обозначается **темно-зеленым** цветом, провод с высоким уровнем – **светло-зеленым** (См рисунок 1). Если что-то нужно будет поменять, не забудьте переключиться в **Режим редактирования**.
- Когда вы получите корректную схему – постарайтесь проверить все возможные состояния, переключая значения входов в **Режим симуляции** и наблюдая за значениями выходов.
- В режиме **Ввода текста** можно наносить произвольный текст на рабочее поле Шрифт и размер текста меняется в меню *Свойства элемента*.

### Часть 3: Разрешенные логические элементы

В дизайне вашей схемы могут присутствовать только **данные элементы** (при наличии других **задание не будет засчитано**):

1. NOT Gate
2. AND Gate
3. NAND Gate
4. OR Gate
5. NOR GATE
6. XOR GATE
7. Probe
8. Pin
9. Соединительные провода

## Часть 4: Варианты задания

Вариант был выслан вам в письме. Делайте схему согласно вашему варианту.

1. Мультиплексор на 8 входов
  2. Компаратор двух 3-битных входов с выходами LEQ (Less Or Equal), EQ (Equal), G (Greater).
  3. **Полный** сумматор (Full Adder) двух 3-битных входов.
  4. D-Flip-Flop
  5. T-Flip-Flop (Входы T, CLK. Может быть сделан через D-Flip-Flop добавлением одного элемента)
- ❖ N-битные входы разрешено оформить в виде N различных проводов на входе

D-Flip-Flop — синоним к D-Триггер, то есть синхронизируется по сигналу CLK. Не путайте с D-Latch. Перепутаете — 0 баллов за все задание.

## Часть 5: Отчет

Отчет должен быть оформлен в виде Word/PDF файла и отправлен на почту [zaitsev.vl@phystech.edu](mailto:zaitsev.vl@phystech.edu) ответным письмом.

**ДЕДЛАЙН 17.03.2025 в 23:59 МСК (через 2 недели).**

### Структура отчета:

1. ФИО
2. Номер группы
3. Название схемы
4. Email почты phystech.edu
5. Назначение схемы. Краткое описание принципов ее работы, ее входов/выходов.

6. **3-4 скриншота** с разными состояниями схемы для последовательной логики и разными входами в случае комбинационной (обязательно учитывая случаи «все нули» и «все единицы» на входе). На схеме должны присутствовать Probe/Pin на входах/выходах. **Входы/выходы должны быть подписаны** (меню ввод текста). **На схеме должно быть нанесено название схемы и фамилия.**
7. **Скриншот** с нарисованным критическим путем в схеме от входа к выходу (красной линией).
8. Примерное число транзисторов в вашей реализации с описанием подсчета. (Вспомните, что  $AND = NAND + NOT$ , которые мы рассмотрели на лекции. XOR также выражается через элементарные функции). При подсчете транзисторов *PROBE* и *PIN* не учитываются.
9. **К письму приложите файл с Logisim схемой (\*.circ)**

**При подозрении на списывание (одинаковые скриншоты схемы) обе работы будут аннулированы.**

**P.S.** Помните, что важна не только сама работа, но и то, как вы ее преподнесите. **Отчет должен СТРОГО соответствовать структуре,** описанной выше, а также быть *эстетически приятным*.

Успехов!