


Установка *DESim* Application

Руководство содержит инструкции по скачиванию и установке приложения *DESim* на компьютер с ОС Microsoft Windows. Для выполнения пунктов, представленных далее, требуется доступ к сети Интернет и необходимые разрешения по установке программного обеспечения.

В данном документе не дается детальных инструкций по использованию приложения *DESim*, а только показывается, как осуществлять его установку. Отдельные инструкции по использованию *DESim* находятся в руководстве *Using the DESim Software with Verilog Code*. Перед использованием этого приложения необходимо установить симулятор *ModelSim*. Инструкции по установке надлежащей версии пакета *ModelSim* даны в руководстве *Using the ModelSim-Intel FPGA Simulator with Verilog Testbenches*, доступном на сайте Intel FPGA Academic Program.

Начало работы

Описанные далее шаги предполагают использование браузера *Google Chrome* для работы в сети Интернет. В случае использования какого-либо другого браузера, возможны небольшие расхождения с инструкцией ниже.

Скачать установщик ПО *DESim* можно с его репозитория на *GitHub*. Откройте свой Интернет-браузер и пройдите по ссылке <https://github.com/fpgacademy/DESim>. Как показано на Рисунке 1, репозиторий *DESim* включает в себя исходный код приложения. Если Вы заинтересованы, то можете просмотреть и его, но это не обязательно. Чтобы загрузить установщик *DESim* на свой компьютер, используйте область *Releases* на странице *GitHub*. Как показано справа на Рисунке 1, щелкните мышью по элементу  Version 1.0 **Latest**. Это действие откроет страницу *Releases* репозитория, часть которой изображена на Рисунке 2.

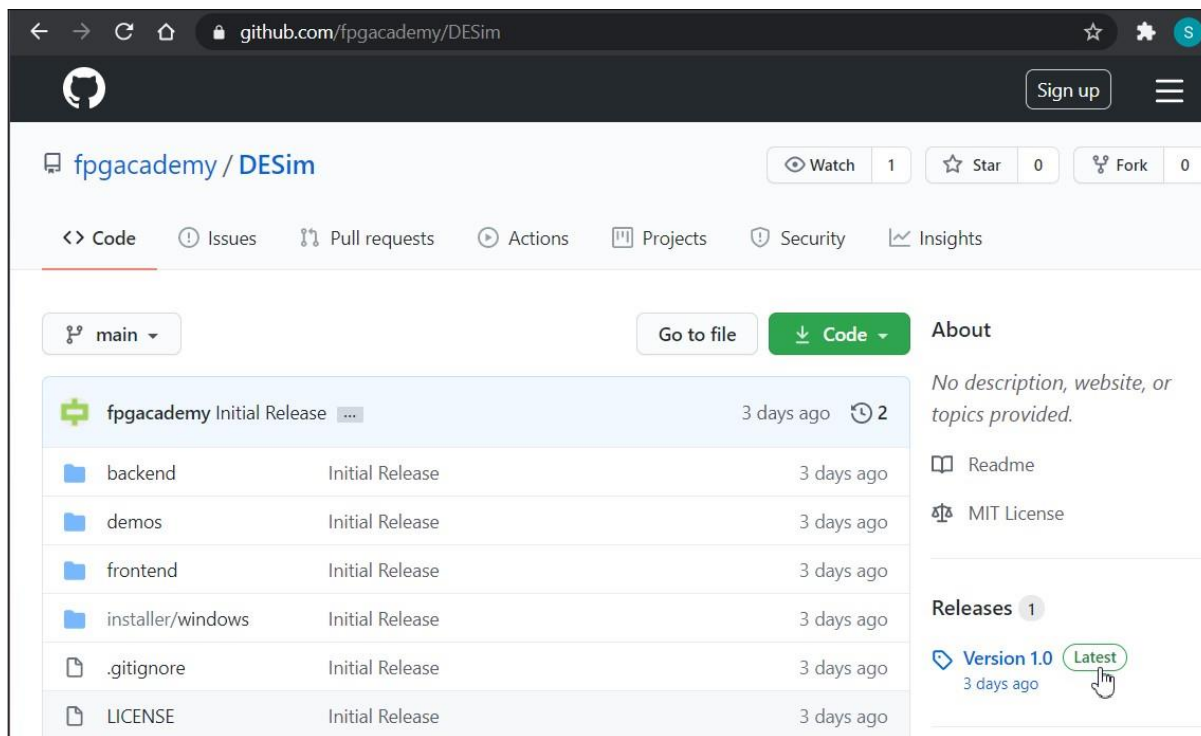


Рисунок 1: Репозиторий *DESim* на *GitHub*.

Кликните по файлу *desim_setup.exe*, как показано на Рисунке 2, что инициализирует загрузку данного файла на Ваш компьютер. В браузере может появиться предупреждающее сообщение, поскольку загружаемый файл является *исполняемой программой*. Выставьте соответствующие настройки в браузере, чтобы сохранить загруженный файл на ПК.

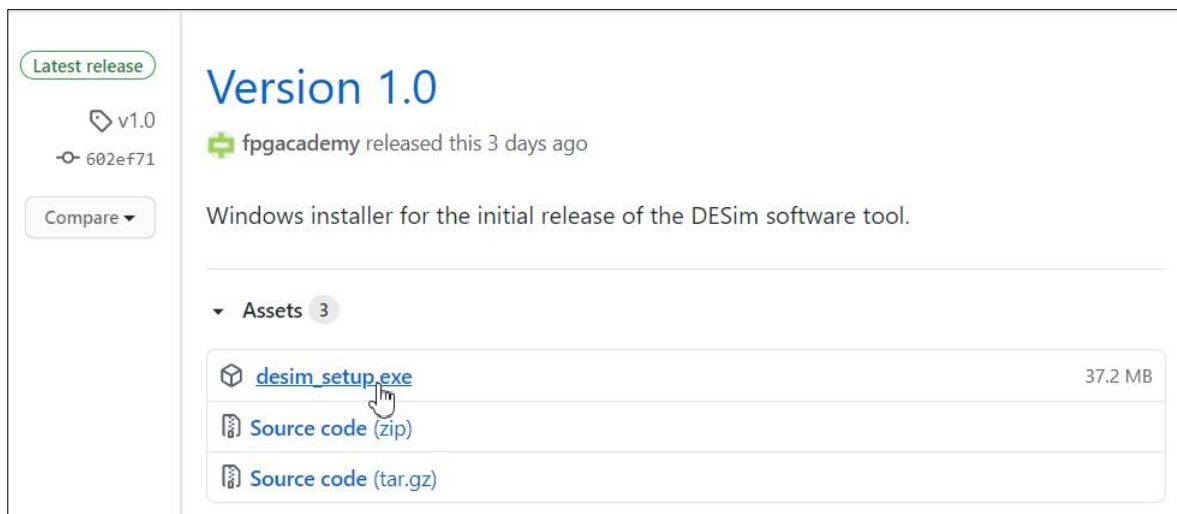


Рисунок 2: Страница Releases в репозитории *DESim*.

Файл *desim_setup.exe* представляет собой *установочную программу* для приложения *DESim*. Откройте этот файл (запустите программу), чтобы перейти к экрану приветствия (Welcome screen), показанному на Рисунке 3. Нажмите Next для просмотра *Лицензионного соглашения* к *DESim*, а затем выберите I Agree, если Вы принимаете условия лицензии. В противном случае программа установки завершит работу. Нажмите Next, чтобы перейти к окну, представленному на Рисунке 4.

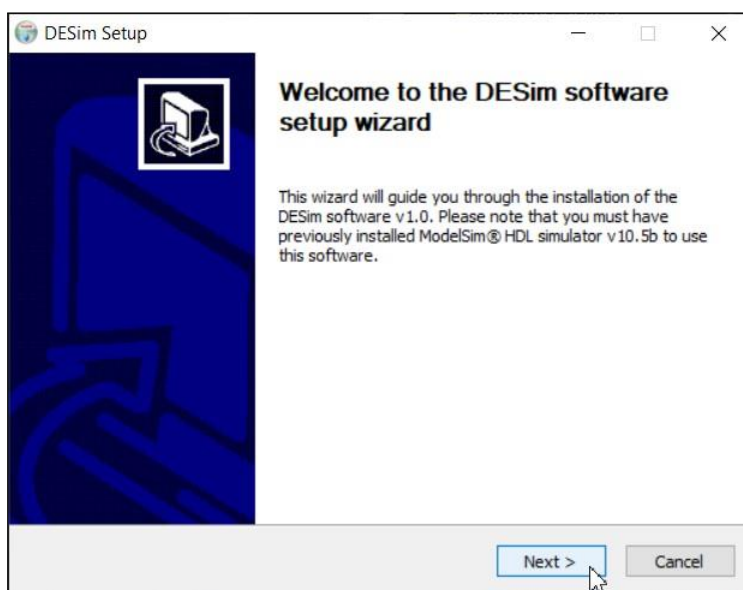


Рисунок 3: Первое окно установщика *DESim*.

В окне на Рисунке 4 Вы можете указать папку для установки. В следующем далее материале предполагается, что было оставлено расположение по умолчанию (C:\DESim), но Вы можете выбрать и другое. Нажмите кнопку Install. В процессе установки есть возможность разместить иконку приложения *DESim* на Рабочем столе, что обеспечит быстрый запуск приложения.

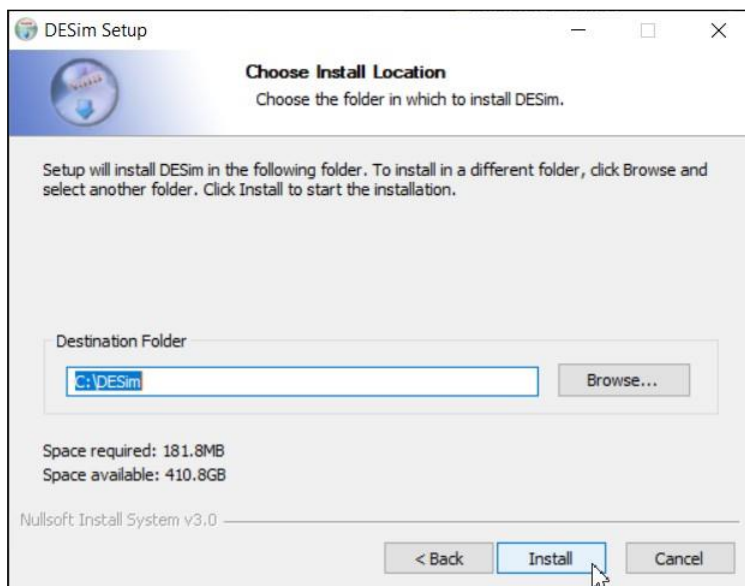


Рисунок 4: Установка приложения *DESim*.

Папка C:\DESim, созданная в процессе установки, содержит программное обеспечение *DESim* и несколько примеров проектов (под названием *demos*). Запустите приложение, дважды щелкнув по его иконке на Рабочем столе, или выбрав *DESim* из меню *Пуск Windows*. Кроме того, Вы можете использовать Проводник, перейдя в папку C:\DESim и щелкнув правой кнопкой мыши по командному файлу *DESim_run.bat*, как показано на Рисунке 5, а затем выбрав *Open* (или просто дважды кликнув по файлу, чтобы сразу его открыть). Теперь Вы должны увидеть графический интерфейс пользователя *DESim* (GUI), проиллюстрированный Рисунком 6. Он должен отображать сообщение “The server is running...” в левой верхней части панели сообщений графического интерфейса.

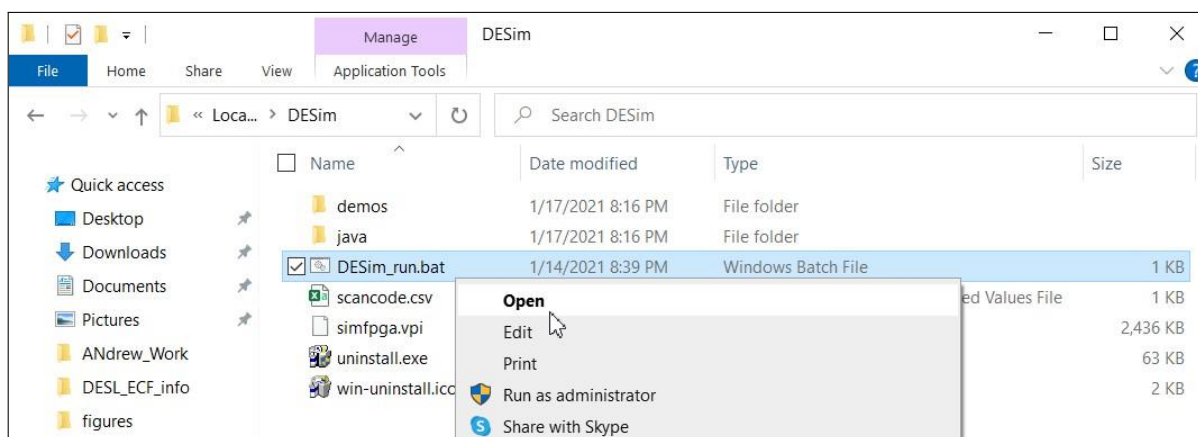


Рисунок 5: Запуск приложения *DESim*.

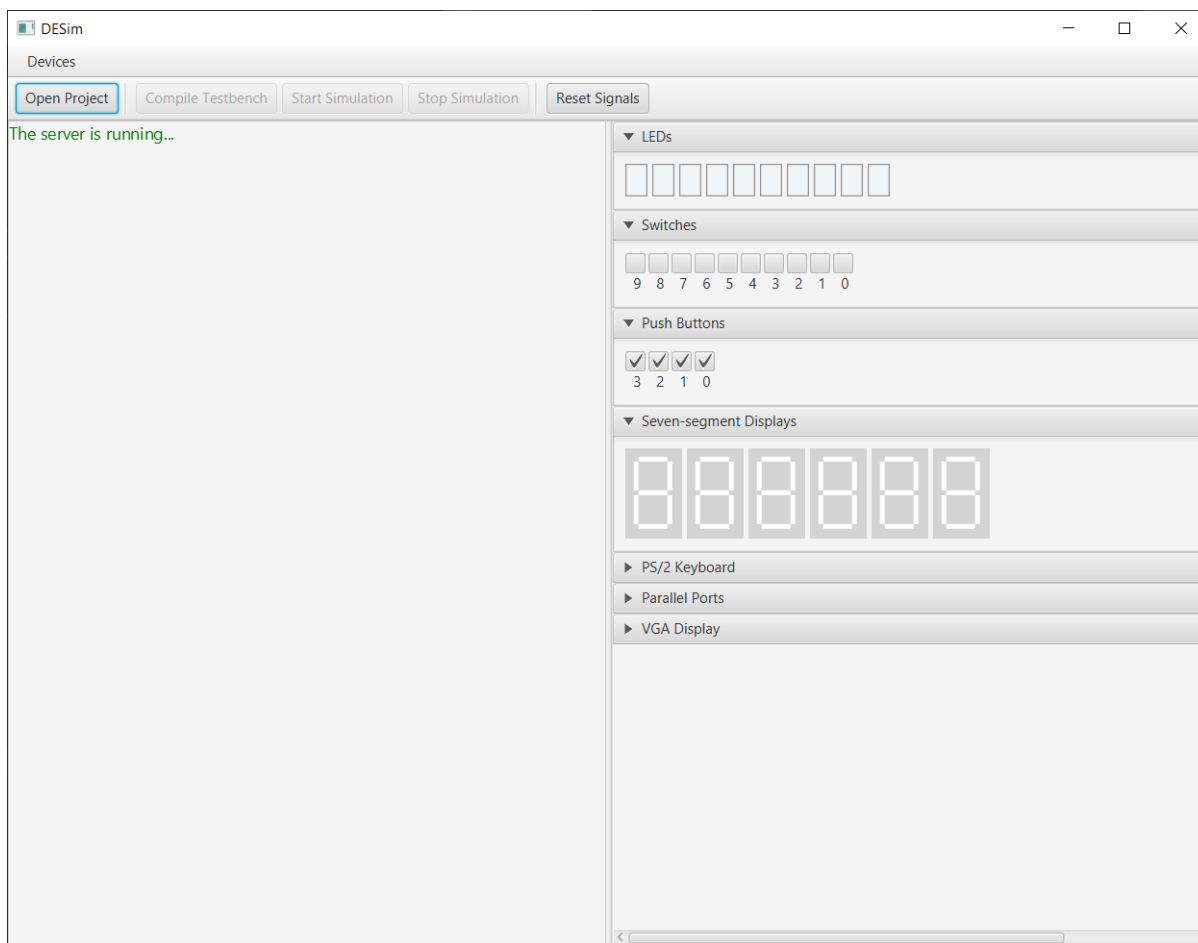


Рисунок 6: Рабочее окно *DESim*.

Чтобы убедиться, что приложение *DESim* может взаимодействовать с симулятором *ModelSim*, Вы можете попробовать запустить один (или более) *demo* проектов, которые прилагаются к *DESim*. А именно, выберите команду `Open Project` в графическом интерфейсе *DESim*, а затем перейдите в папку `demos`. Как показано на Рисунке 7, щелкните по папке `LED_HEX` и потом нажмите на кнопку `Select Folder`.

Выберите команду `Compile Testbench`. Как отражено на Рисунке 8, симулятор *ModelSim* используется для компиляции кода на языке Verilog, а сообщения о компиляции, созданные *симулятором*, отображаются в панели сообщений *DESim*.

В окне *DESim* выберите команду `Start Simulation`, которая запускает симулятор *ModelSim Verilog simulator*. Как отражено на Рисунке 9, сгенерированные *ModelSim* сообщения отображаются в панели сообщений окна *DESim*. Чтобы привести дисплей к виду, как на рисунке, щелкните в графическом интерфейсе *DESim* по переключателю (Switch) под номером 6, что приведет к загоранию соответствующего светодиода (LED) **красным**. Для активации вывода семисегментного дисплея (Seven-segment Display) нужно произвести сброс сигналов. Для этого сначала щелкните по Push Button 0, чтобы нажать ее, а затем щелкните еще раз, чтобы отжать. Чтобы подробнее ознакомиться с данным демо-проектом `LED_HEX`, Вы можете изучить идущий к нему текстовый файл *Readme.txt*, показанный на Рисунке 10, и/или прочитать файл с исходным кодом на языке Verilog `LED_HEX.v`, изображенный на Рисунке 11. Эти файлы можно найти с помощью Проводника Microsoft Windows в папке `LED_HEX`.

Вы можете остановить симуляцию *ModelSim*, выбрав команду `Stop Simulation` в *DESim* GUI. А чтобы закрыть программу *DESim*, щелкните значок `X` в правом верхнем углу окна.

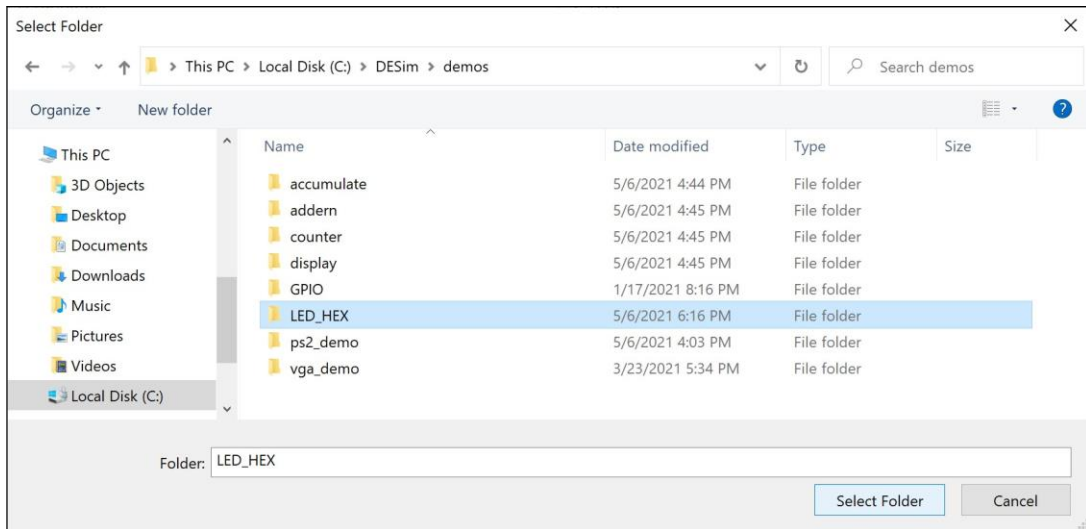


Рисунок 7: Открытие образца проекта в папке demos.

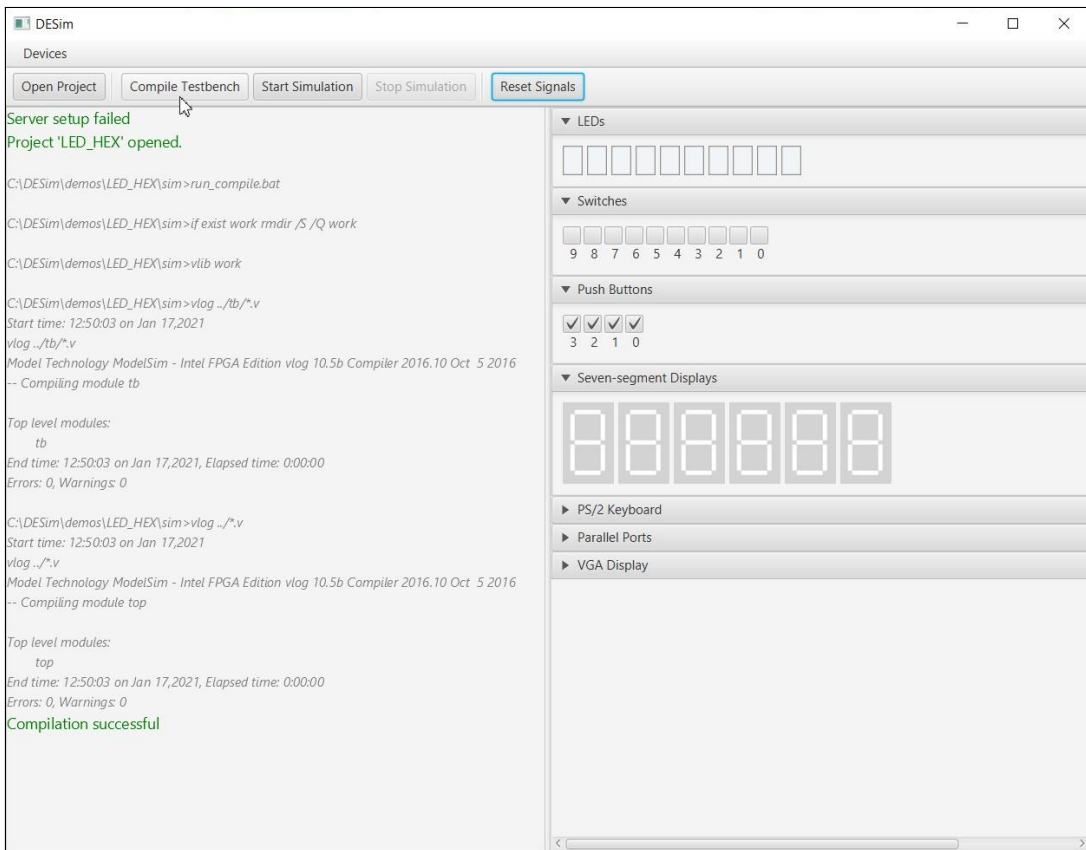


Рисунок 8: Компилирование образца проекта.

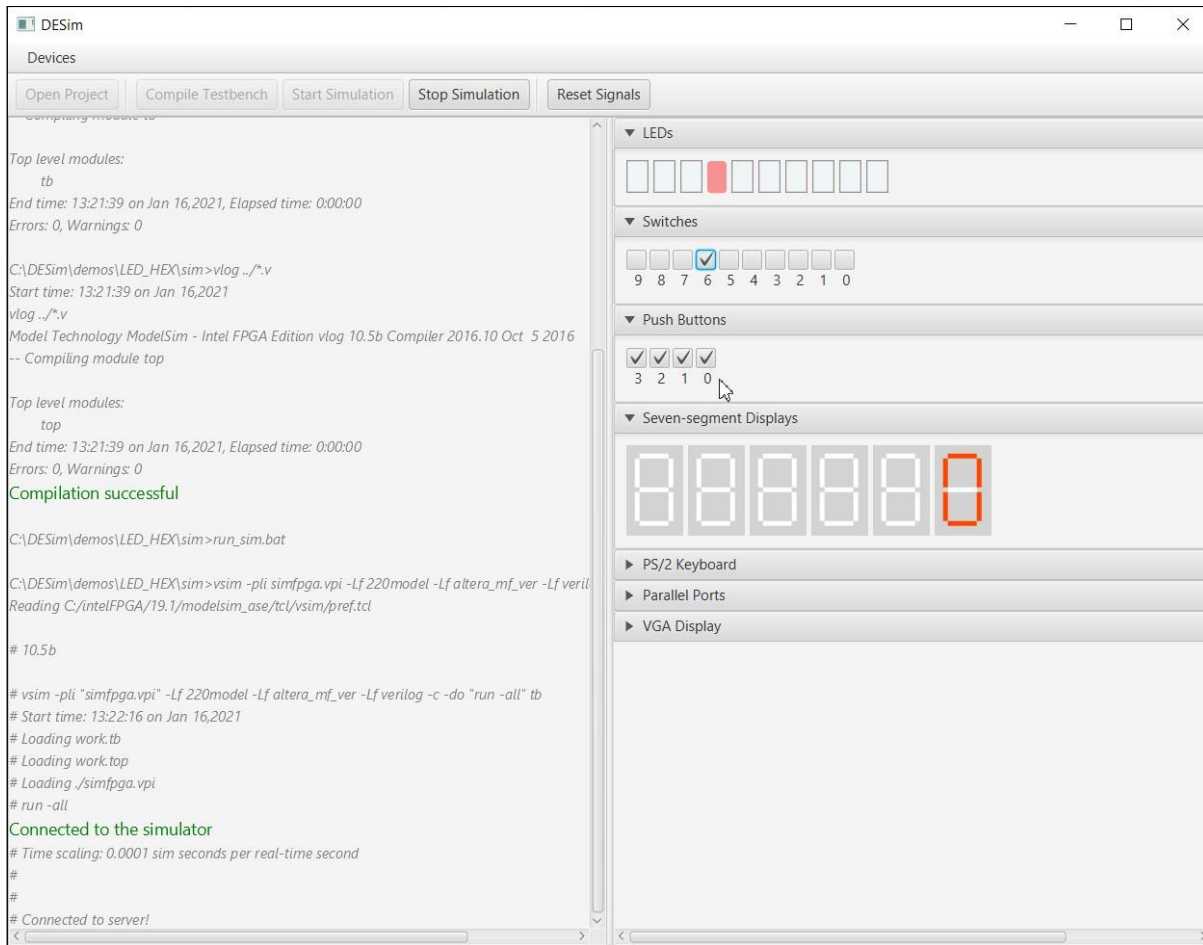


Рисунок 9: Симуляция образца проекта.

To use this demo:

- SW are displayed on LEDR
- KEY[0] is the synchronous reset. It sets the HEX-display selector to 0.
- KEY[1] provides the active-low enable for the HEX-display selector

To use:

1. First press/release KEY[0] to reset the circuit; HEX0 is selected
 - HEX0 can be changed using SW[6:0]
2. Set SW[9:7] to select a different HEX display (from 0 to 5)
 - press/release KEY[1] to store the selected HEX address
 - the selected HEX display can now be changed using SW[6:0]
3. Set SW[9:7] to select another display, etc.

Рисунок 10: Файл *Readme.txt*, прилагающийся к проекту LED_HEX.

```

module LED_HEX (CLOCK_50, SW, KEY, LEDR, HEX0, HEX1, HEX2, HEX3, HEX4, HEX5);
    input wire CLOCK_50;
    input wire [9:0] SW;
    input wire [1:0] KEY;
    output wire [9:0] LEDR;           // DE-series LEDs

    output reg [6:0] HEX0;           // DE-series HEX displays
    output reg [6:0] HEX1;
    output reg [6:0] HEX2;
    output reg [6:0] HEX3;
    output reg [6:0] HEX4;
    output reg [6:0] HEX5;

    assign LEDR = SW;

    reg [2:0] Addr;                  // used to select a HEX display

    always @ (posedge CLOCK_50)
        if (KEY[0] == 0)             // sync reset
            Addr <= 3'b0;
        else if (KEY[1] == 0)        // select a HEX display
            Addr <= SW[9:7];

    always @ (posedge CLOCK_50)
        case (Addr)
            3'b000:  HEX0 <= SW[6:0];
            3'b001:  HEX1 <= SW[6:0];
            3'b010:  HEX2 <= SW[6:0];
            3'b011:  HEX3 <= SW[6:0];
            3'b100:  HEX4 <= SW[6:0];
            3'b101:  HEX5 <= SW[6:0];
            default: ;
        endcase
endmodule

```

Рисунок 11: Исходный код Verilog-файла LED_HEX.v.