

Quick Start 5400TP194

1	Установка.....	2
1.	Online.....	2
2.	Offline.....	2
2	Запуск программы Eclipse.....	3
3	Создание проекта.....	5
4	Смена перспективы.....	12
5	Сборка проекта.....	13
6	Моделирование.....	14
7	Дальнейшая работа с проектом QuickStart.....	20

1 Установка

1. Online

1. Разархивировать в любую удобную директорию архив dcs-emulator.tar.gz (например в /home/<username>/opt).
<username> - имя пользователя
2. Разархивировать в любую удобную директорию архивы dcs-eclipse.tar.gz и risc-v-toolchain.tar.gz (например в /home/<username>/opt)
3. Добавить ссылку на репозиторий ДЦ Союз в свой ПК запуском скрипта install_user_dcs_repo.sh:

```
1 sudo bash <path_to_script>/install_user_dcs_repo.sh
```

*<path_to_script> - путь до директории, в которой находится
install_user_dcs_repo.sh*

4. Обновить список доступных пакетов программного обеспечения и установить необходимые пакеты, используя следующие команды в терминале Linux:

```
1 sudo apt update
```

```
2 sudo apt install scemu-mcu5400tp194 scemu-analog scemu-digital openjdk-17-jdk make gtwave -y
```

*Примечание: команды могут выполняться некоторое время, необходимо
дождаться завершения.*

2. Offline

1. Разархивировать в любую удобную директорию архив dcs-<packages-names>-<dist>.tar.gz (например в /home/<username>/opt).
<packages-names> - имена пакетов, которые содержатся в сборке
<username> - имя пользователя
<dist> - дистрибутив Linux (Ubuntu 22.04 — jammy, Debian 12 — bookworm)
2. Перейти в директорию dcs-<packages-names>-<dist>, которая расположена по пути ранее выбранному для разархивирования dcs-<packages-names>-<dist>.tar.gz.

3. Запустить скрипт `install-<packages-names>.sh` из директории `dcs-<packages-names>-<dist>`, выполнив в терминале следующую команду

```
1 bash install-<packages-names>.sh.
```

Примечание: во время выполнения скрипта необходимо ввести пароль от sudo.

В случае успешного выполнения скрипта, `dcs-eclipse` будет находиться по пути `/home/<username>/opt/dcs-eclipse`, `risc-v-toolchain` будет находиться по пути `/home/<username>/opt/RISC-V_Embedded_GCC12`.

2 Запуск программы Eclipse

Для запуска программы Eclipse необходимо:

1. Перейти в директорию `dcs-eclipse` и дважды нажать на `eclipse` (Рис. 1)

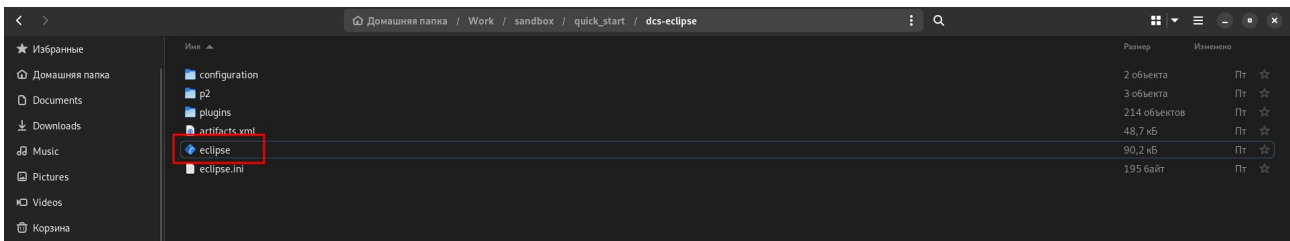


Рисунок 1 - Программа eclipse

2. Выбрать расположение workspace нажатием на кнопку `Browse` (Рис. 2) и последующим выбором директории

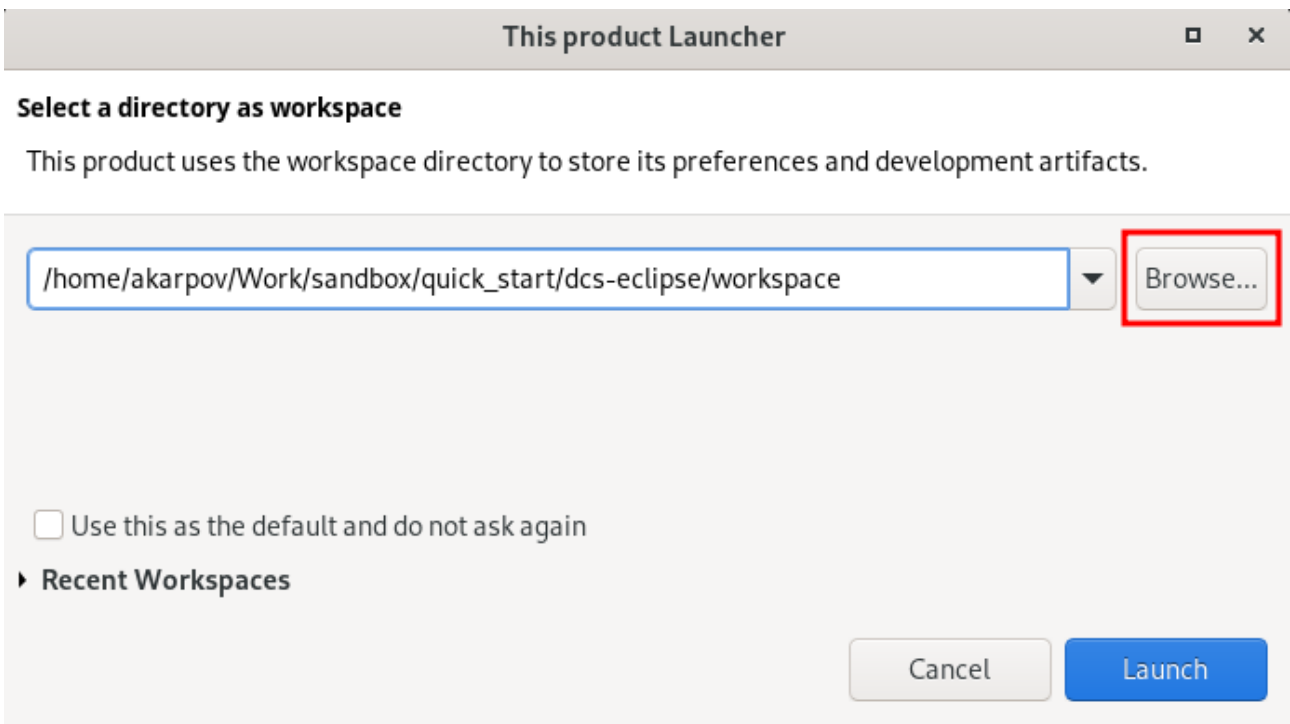


Рисунок 2 - Выбор workspace

3. Нажать на кнопку Launch (Рис. 3)

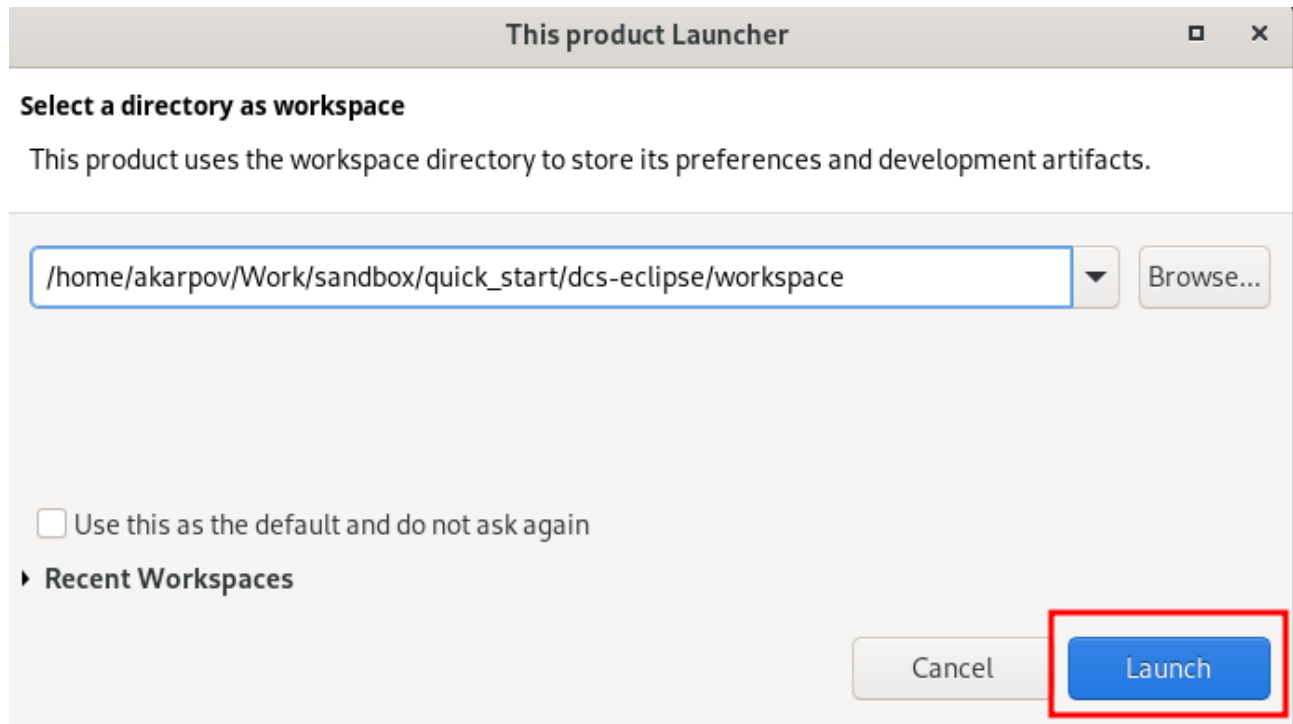


Рисунок 3 - Launch

3 Создание проекта

Для создания проекта Quick Start необходимо:

1. Нажать на кнопку «Create a project» в окне «Project Explorer», расположенном в левой части интерфейса программы (Рис. 4)

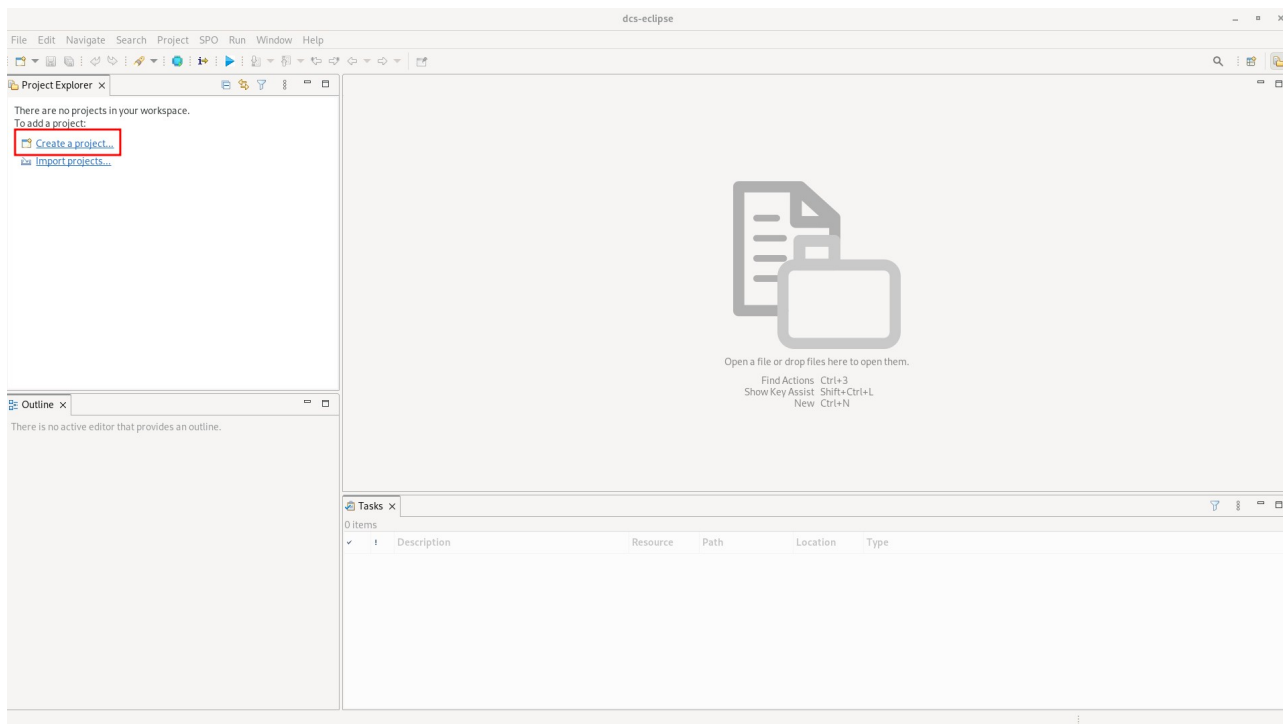


Рисунок 4 - Create a project

2. Раскрыть меню «C/C++» и дважды нажать на кнопку «C/C++ Project» (Рис. 5)

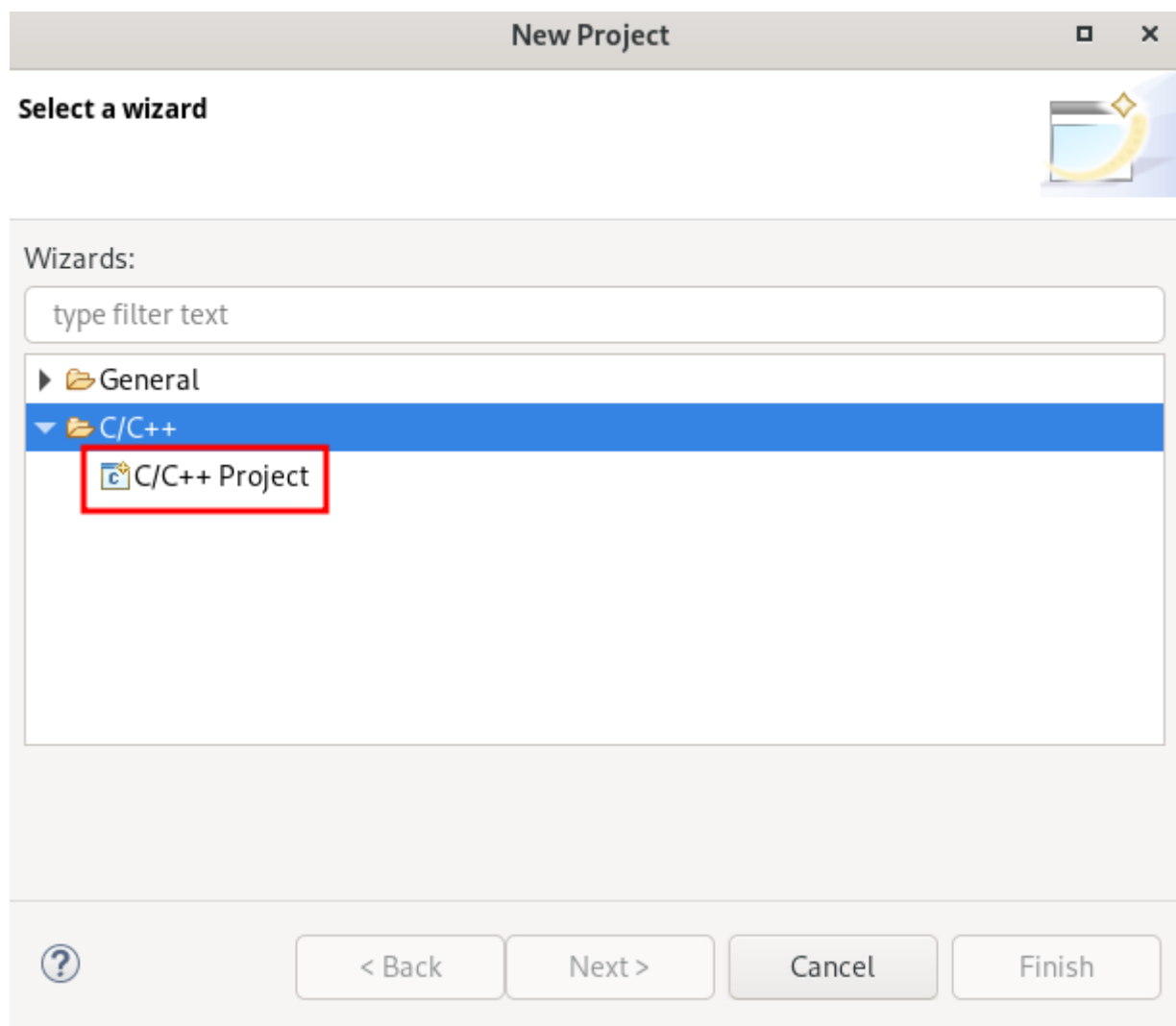


Рисунок 5 - C/C++ Project

3. Нажать на кнопку «C++ Managed Build» (Рис. 6)
4. Нажать на кнопку «Next» (Рис. 6)

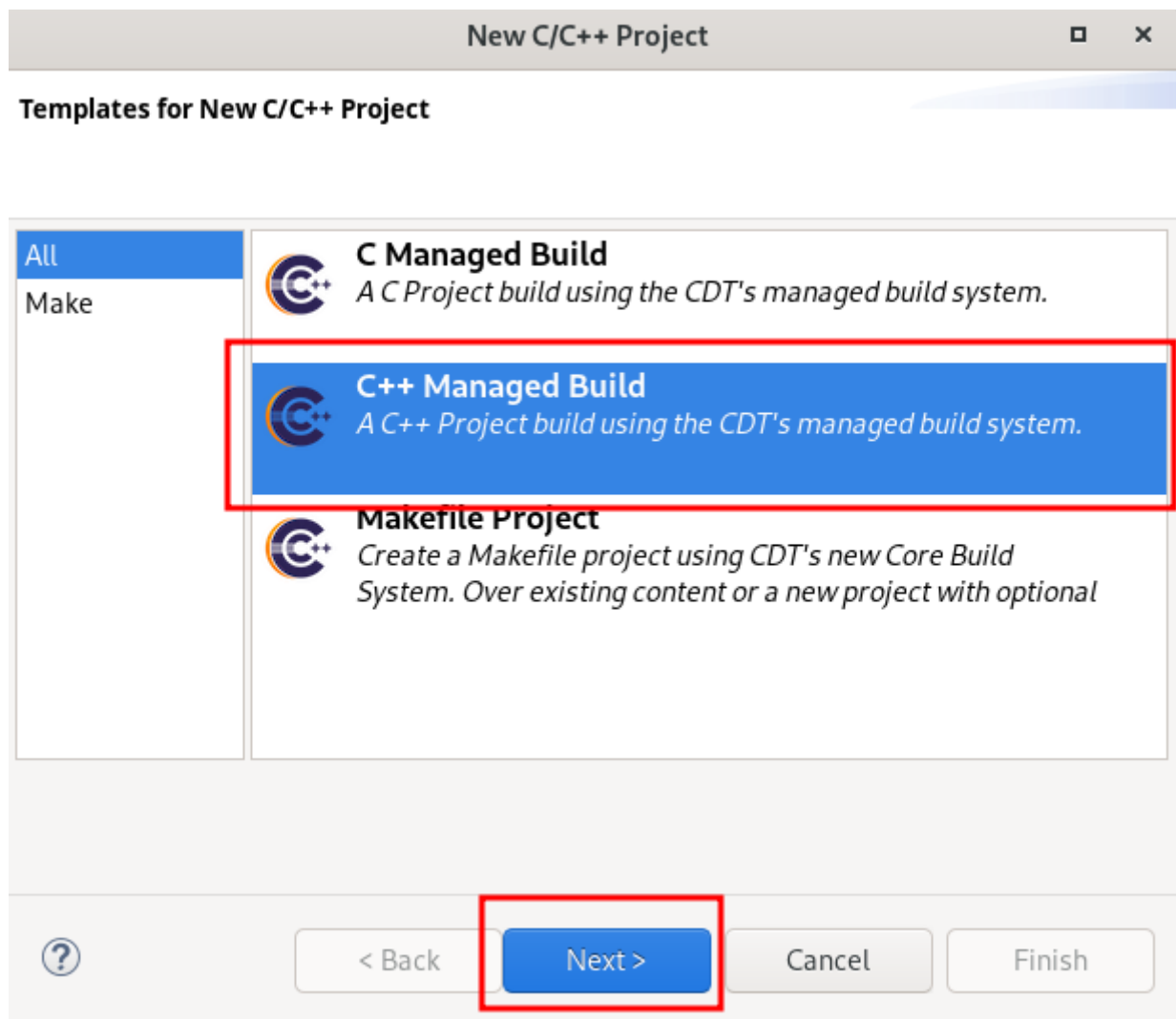


Рисунок 6 - C++ Managed Build

5. В появившемся окне в поле «Project Name» написать желаемое название проекта, выбрать «5400TP194 QuickStart C++ Project» и нажать кнопку «Next» (Рис. 7)

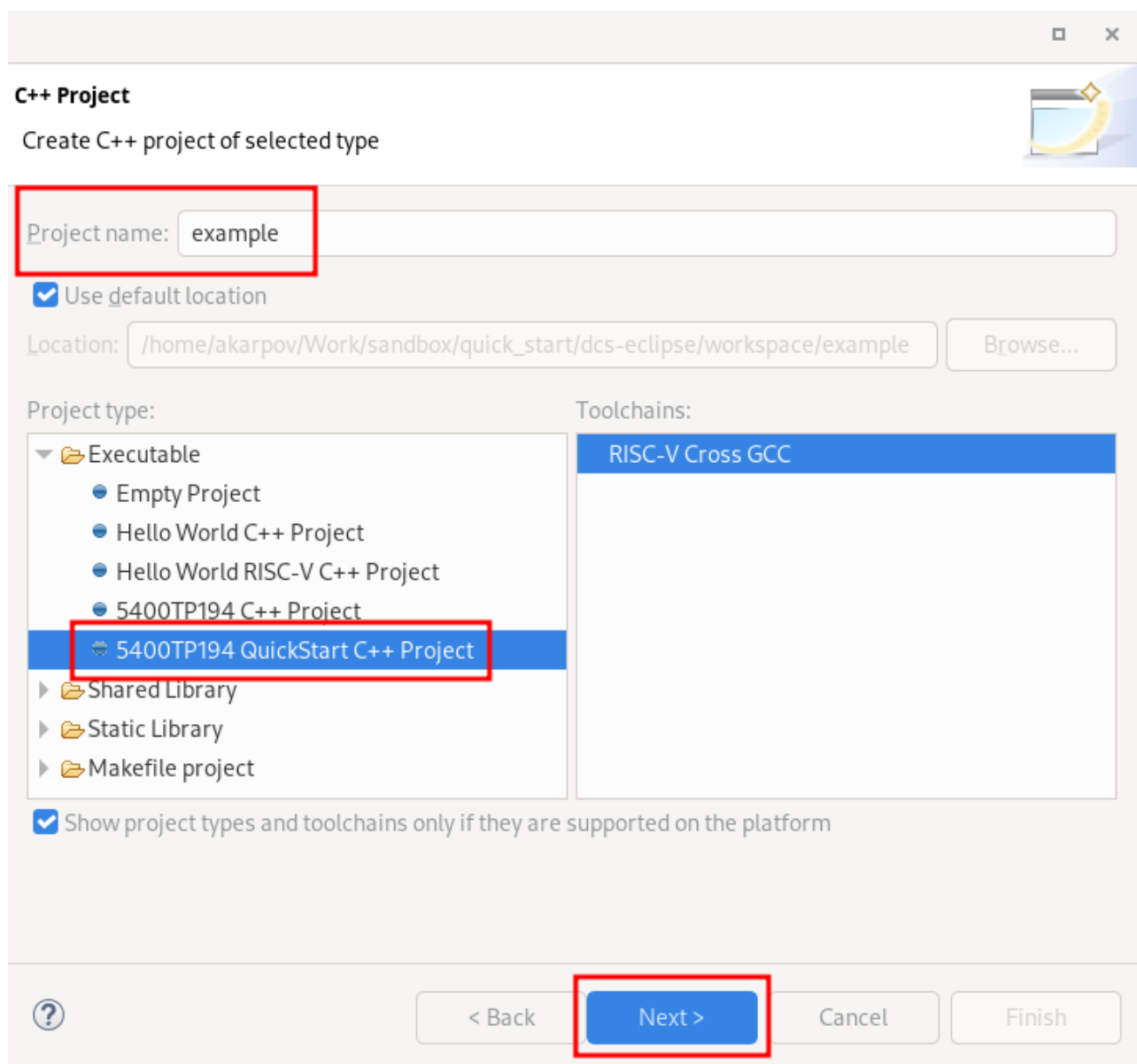
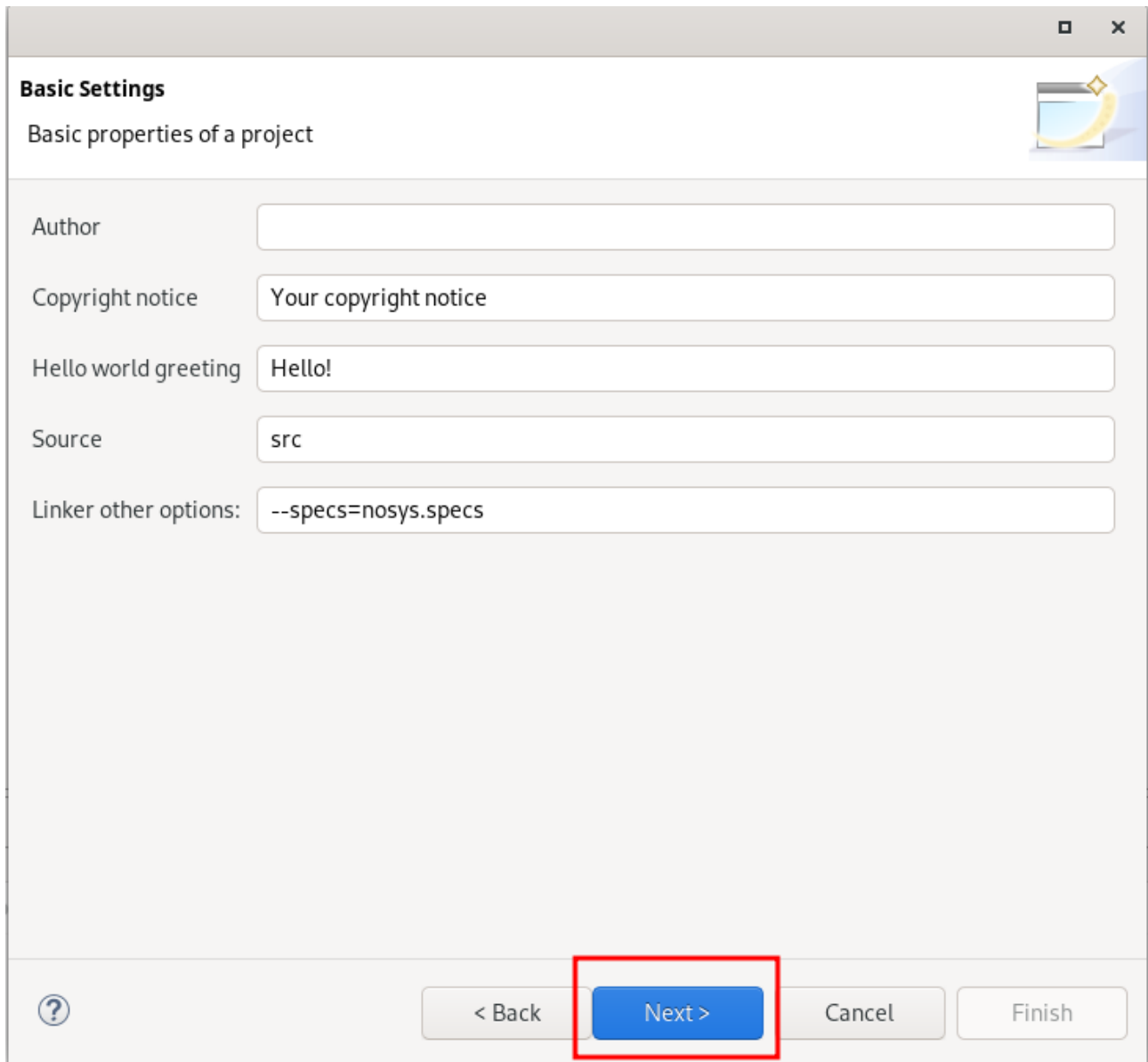


Рисунок 7 - Создание проекта

6. Нажать кнопку «Next» (Рис. 8)



Basic Settings
Basic properties of a project

Author

Copyright notice

Hello world greeting

Source

Linker other options:



Рисунок 8 - Создание проекта

7. Нажать кнопку «Next» (Рис. 9)

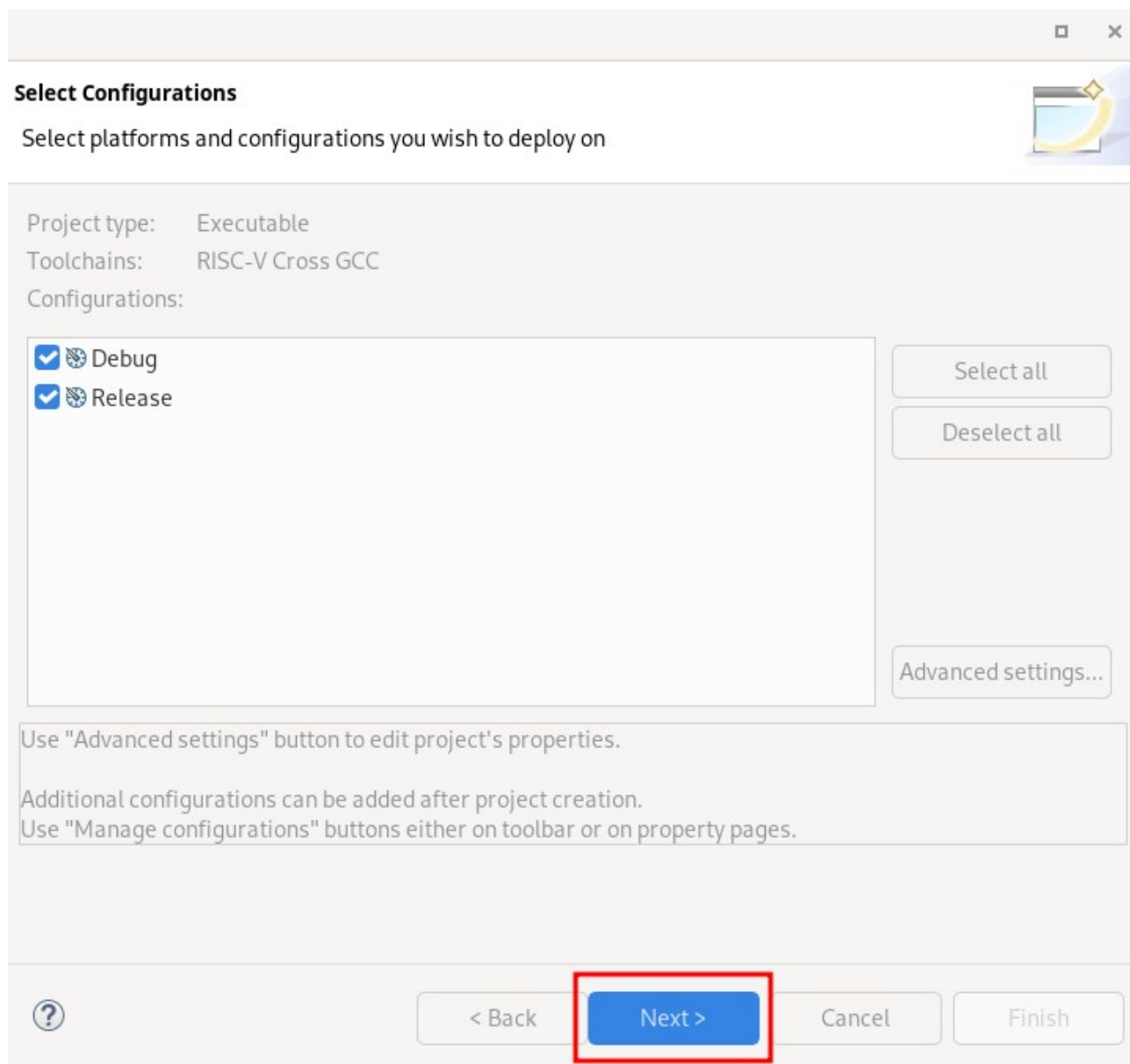


Рисунок 9 - Создание проекта

8. Нажать кнопку «Browse», выбрать путь до toolchain (в данном случае RISC-V_Embedded_GCC12) глубиной включая директорию «bin» и нажать кнопку «Finish» (Рис. 10)

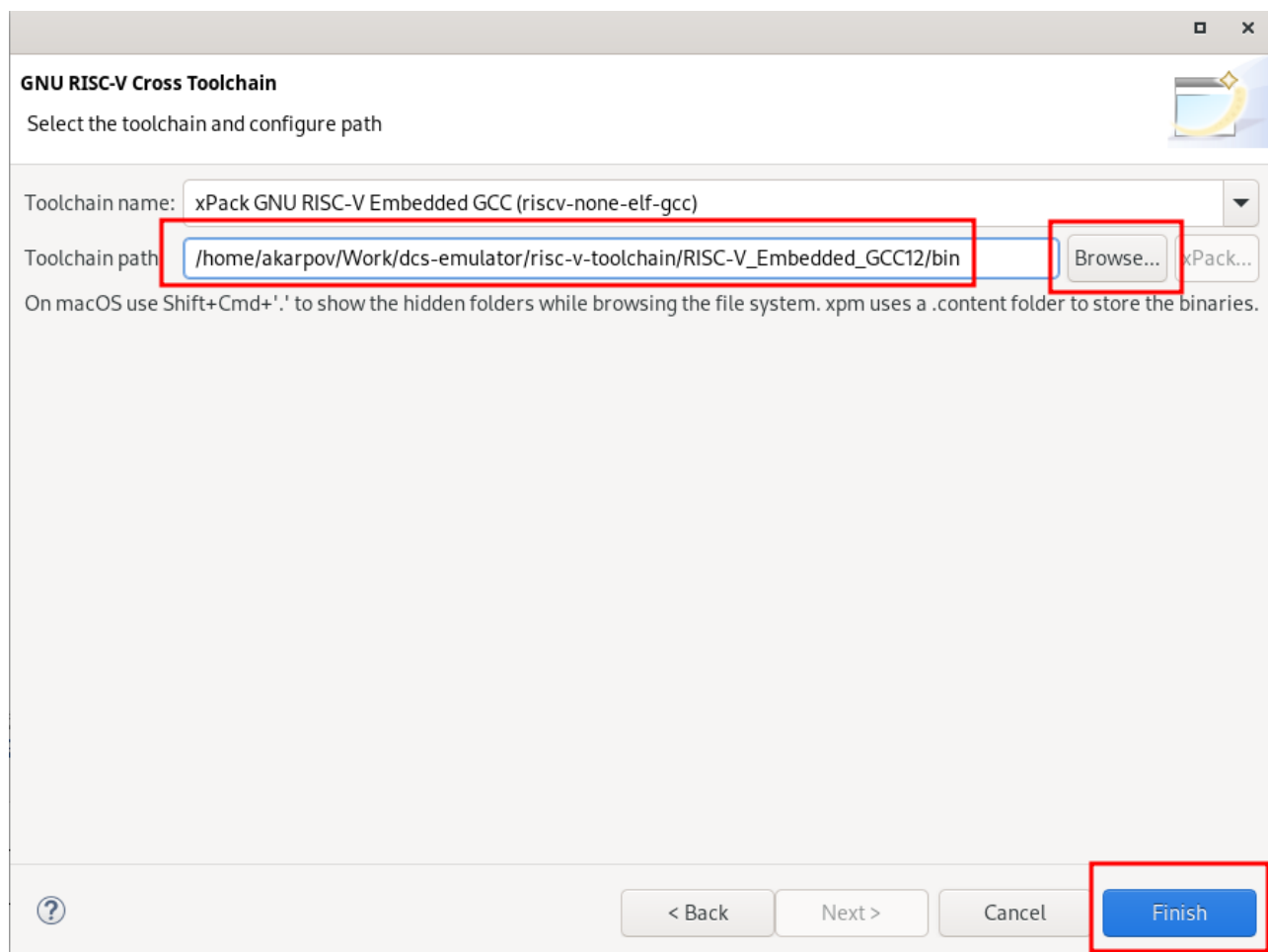


Рисунок 10 - Выбор toolchain

4 Смена перспективы

Для смены перспективы необходимо нажать на кнопку «Open Perspective» в правом верхнем углу (Рис. 11), выбрать «C/C++» и нажать кнопку «Open» (Рис. 12)

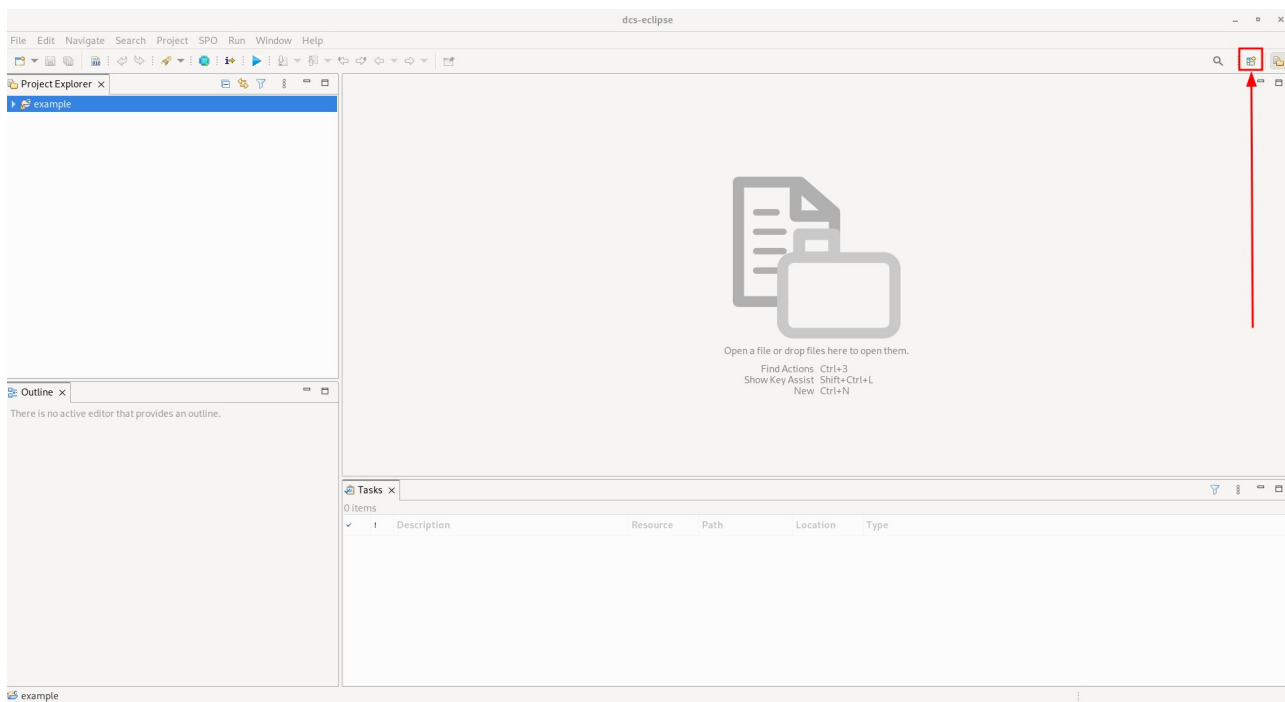


Рисунок 11 - Open Perspective

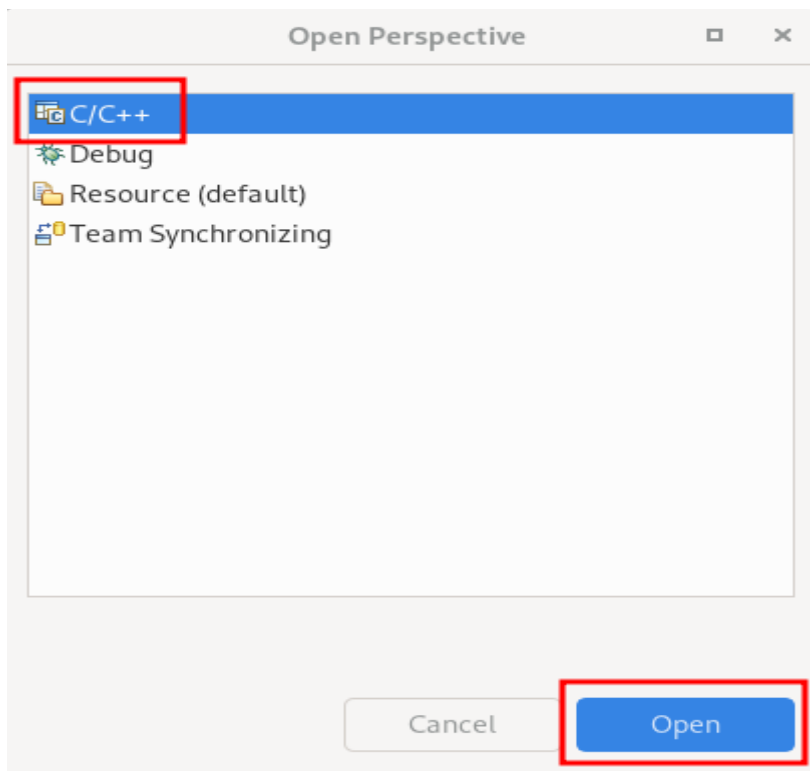


Рисунок 12 - Open Perspective

5 Сборка проекта

Для сборки проекта необходимо нажатием выбрать ранее созданный проект в окне «Project Explorer» в левой части интерфейса программы (Рис. 13), нажать на стрелочку рядом с кнопкой «Build» и нажать на кнопку «Debug» (Рис. 14)

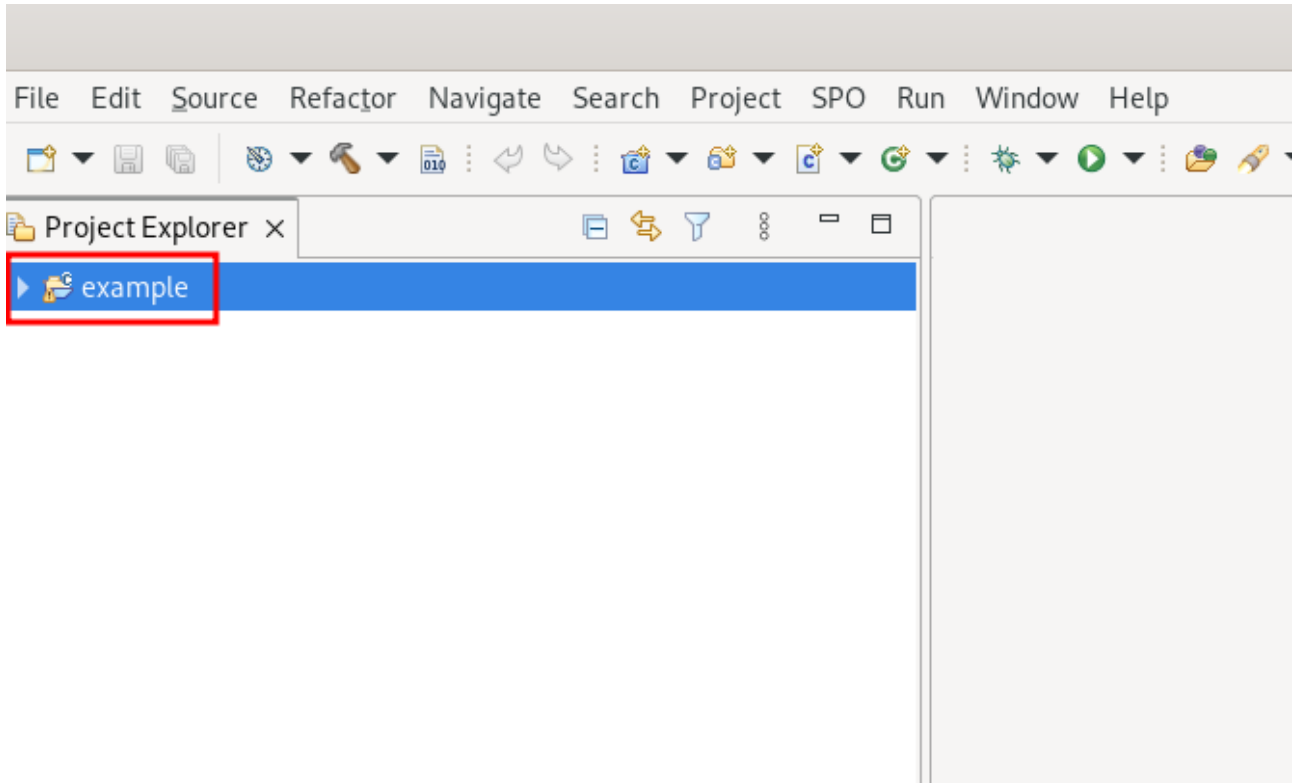


Рисунок 13 - Выбор проекта

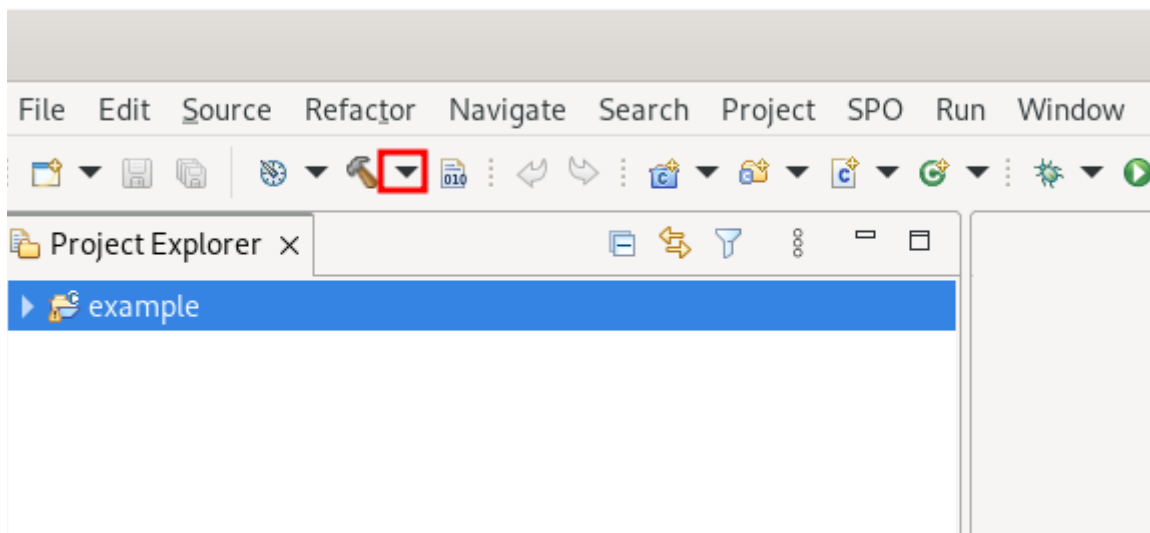
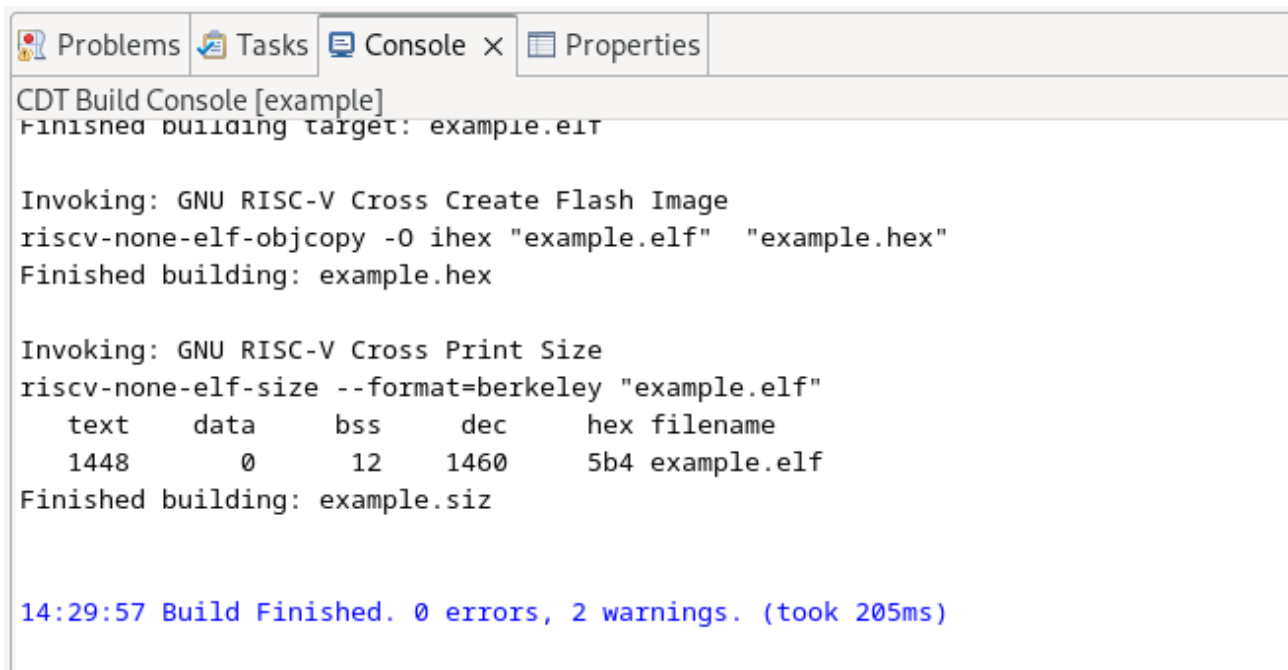


Рисунок 14 - Build

В случае успешной сборки проекта в нижней части интерфейса в окне «Console» выведется соответствующий лог (Рис. 15)



```
CDT Build Console [example]
Finished building target: example.elf

Invoking: GNU RISC-V Cross Create Flash Image
riscv-none-elf-objcopy -O ihex "example.elf" "example.hex"
Finished building: example.hex

Invoking: GNU RISC-V Cross Print Size
riscv-none-elf-size --format=berkeley "example.elf"
  text    data    bss    dec    hex filename
  1448      0     12   1460   5b4 example.elf
Finished building: example.siz

14:29:57 Build Finished. 0 errors, 2 warnings. (took 205ms)
```

Рисунок 15 - Лог сборки проекта

6 Моделирование

Для моделирования необходимо:

1. Раскрыть проект нажатием на кнопку раскрытия дерева проекта (Рис. 16)

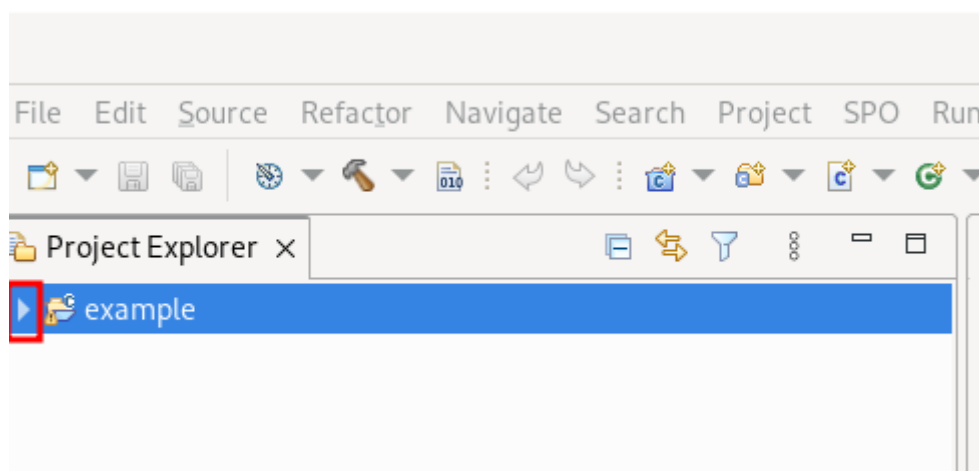


Рисунок 16 - Кнопка раскрытия дерева проекта

2. Открыть файл «config.json» последовательно нажав ПКМ на файл и выбрав «Open With → Text Editor» (Рис. 17)

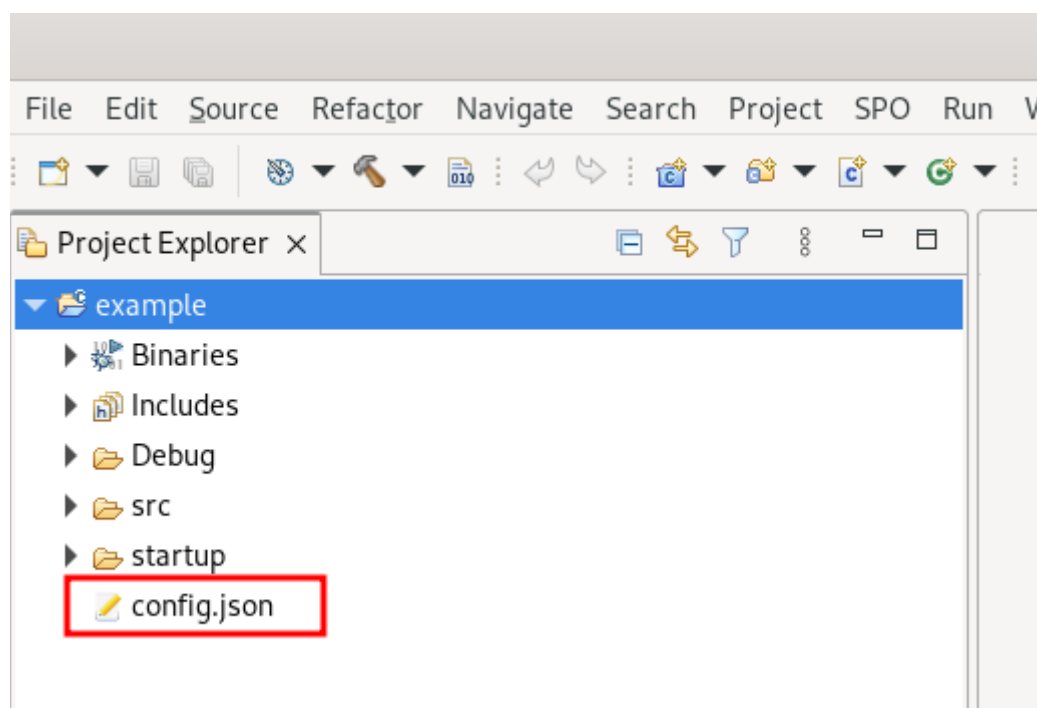


Рисунок 17 - config.json

3. В блоке «m5400tr194» в поле «program» указать путь до hex файла (Рис. 18), полученного после сборки проекта (Рис. 19), имя которого определяется названием проекта (например, example — example.hex, test_project — test_project.hex)

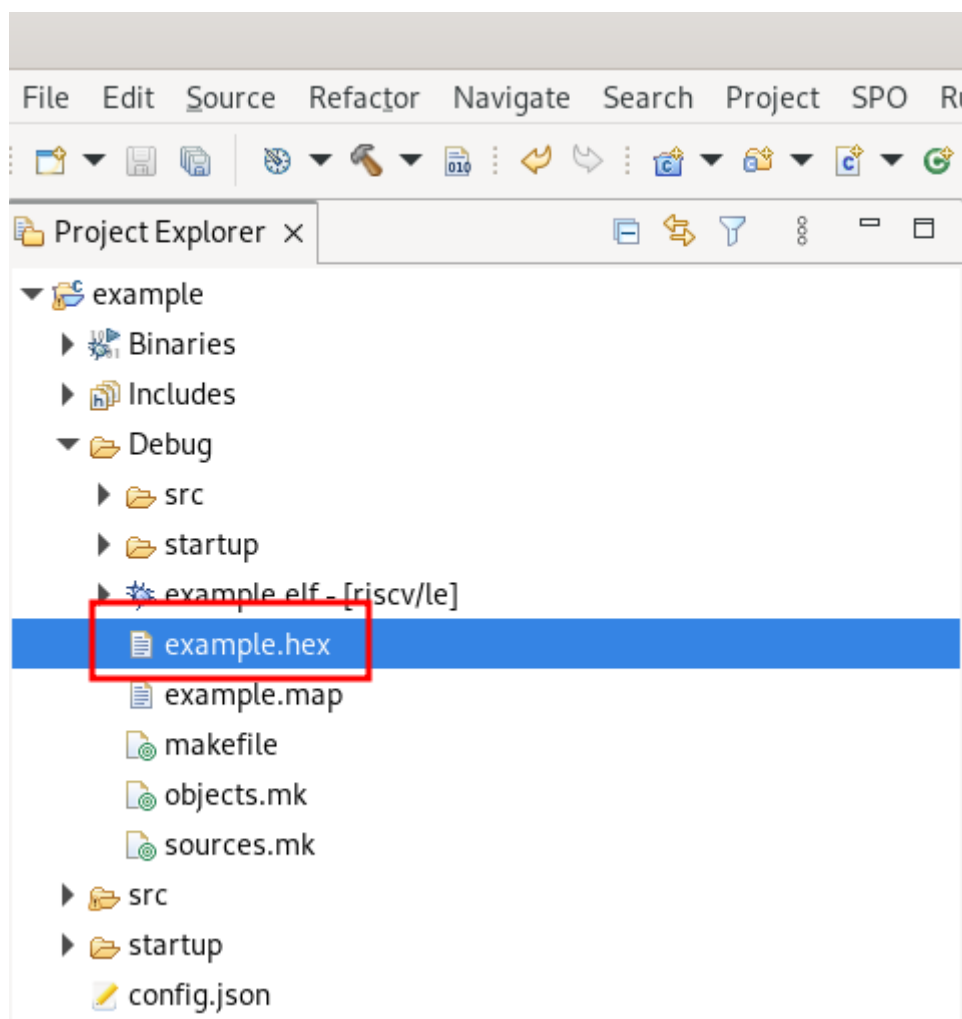


Рисунок 18 - example.hex

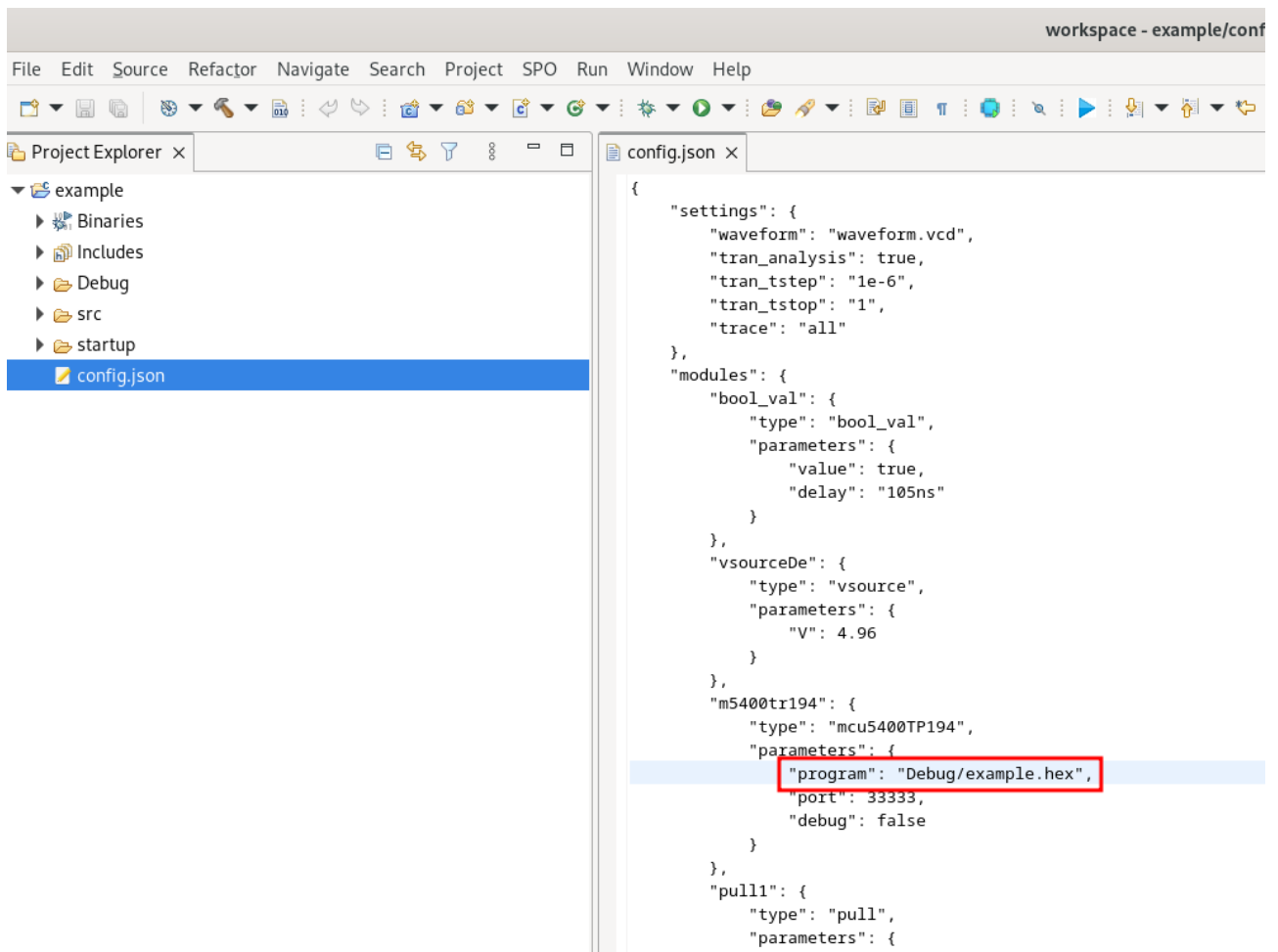


Рисунок 19 - config.json

4. Сохранить файл config.json нажав комбинацию клавиш Ctrl+S

5. Нажать на кнопку «emustart» расположенную в верхней части интерфейса программы (Рис. 20)

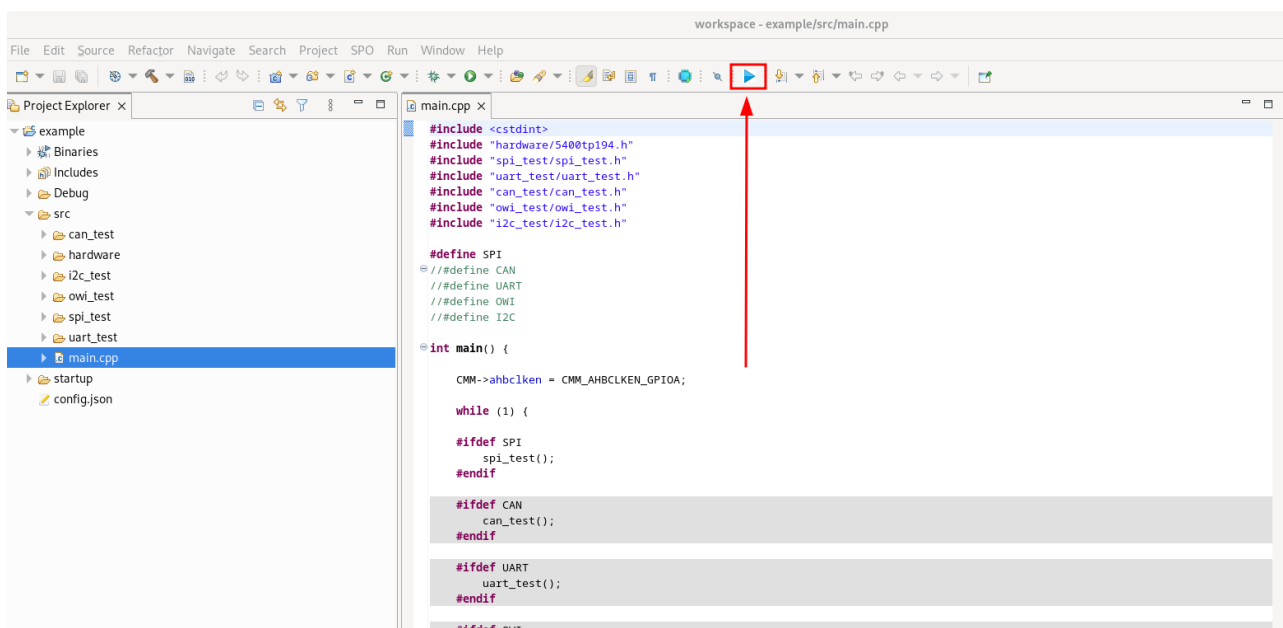


Рисунок 20 - emustart

6. Дождаться окончания моделирования, нажать на имя проекта в окне «Project Explorer» и нажать клавишу F5, в корне проекта должен появиться файл «waveform.vcd» (Рис. 21)

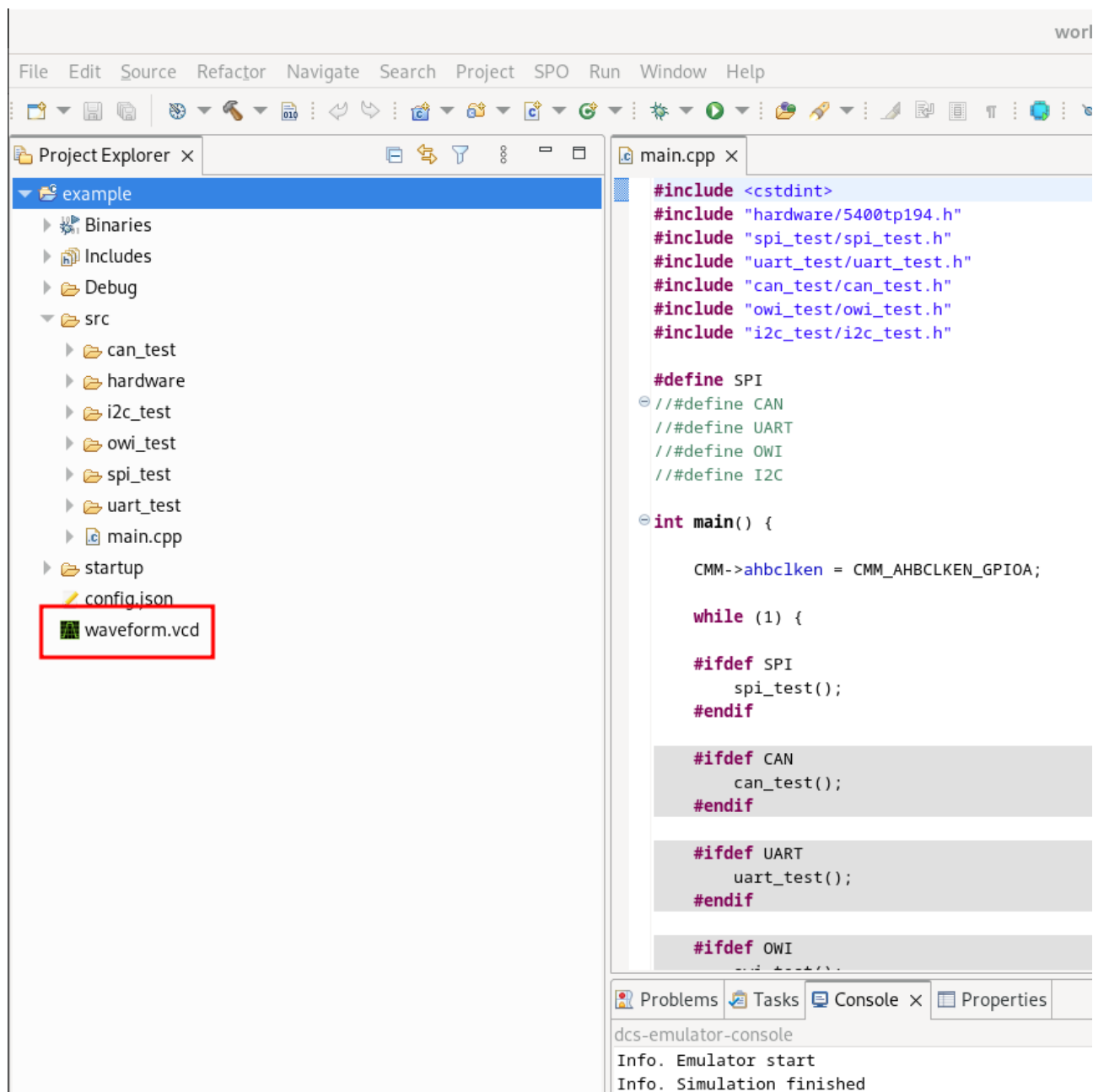


Рисунок 21 - waveform.vcd

7 Дальнейшая работа с проектом QuickStart

В проекте представлены функции, реализующие программирование следующих интерфейсов: SPI, CAN, UART, OWI, I2C. Для удобства вызов функций реализован через макрос `#define`. Для запуска теста того или иного интерфейса необходимо раскомментировать строчку с макросом `#define` и названием целевого интерфейса в файле `main.cpp` (Рис. 22), строчки с названиями остальных интерфейсов — закомментировать (Рис. 23), провести повторную сборку проекта и запустить моделирование (без изменения поля «program» в файле `config.json`).

Примечание: перед запуском моделирования необходимо проверить сохранены ли изменения в файле `main.cpp`.

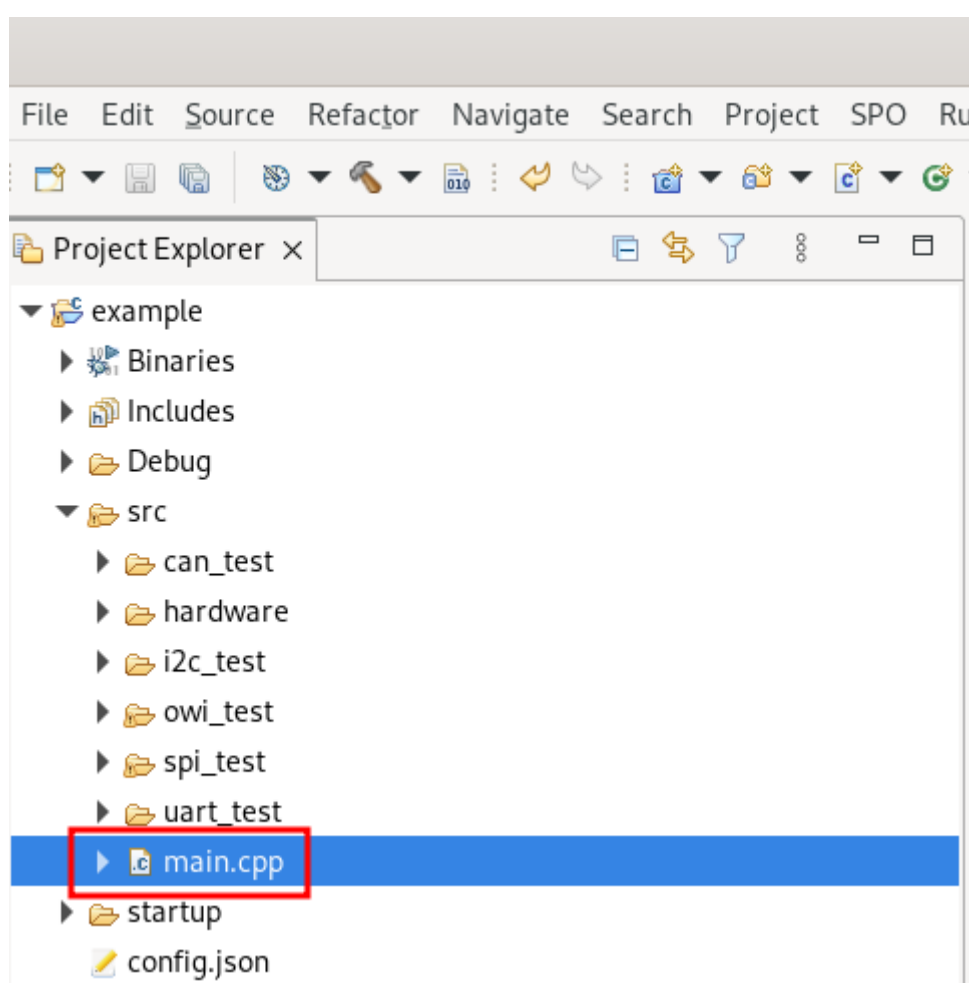


Рисунок 22 - main.cpp

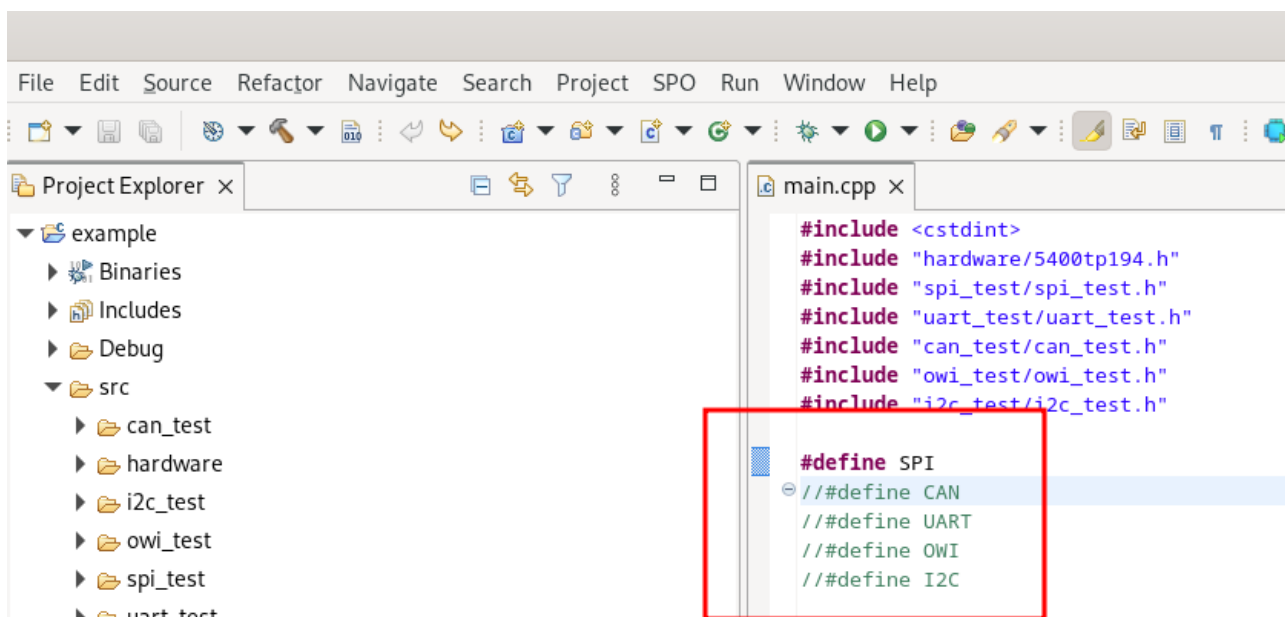


Рисунок 23 - Defines

За более подробным описанием эмулятора микросхемы 5400TP194 обращайтесь в документ 5400tp194-emulator.pdf.