

Настройка окружения в Ubuntu 20.04 для работы с OpenACC

Оглавление

Настройка окружения в Ubuntu 20.04 для работы с OpenACC.....	1
Установка компиляторов в Ubuntu 20.04	1
Development Tools packages (gcc, g++, make, clang).....	1
Установка NVIDIA HPC SDK 21.3	2
Установка программного окружения OpenGL и GLUT	3
Настройка Code::Blocks.....	3
Настройка Code::Blocks для использования установленных компиляторов	3
GNU GCC 10 Compiler.....	3
Clang 11	6
Nvidia nvc++ 21.3-0 LLVM	8
Настройка опций сборки в Code::Blocks для OpenMP и OpenACC с использованием установленных компиляторов.....	10
GCC 9 и GCC 10	11
Clang 10 и Clang 11	15
NVIDIA HPC Compiler nvc++ 21.3-0 LLVM	17

Установка компиляторов в Ubuntu 20.04

Development Tools packages (gcc, g++, make, clang)

```
$ sudo apt update
```

```
$ sudo apt install build-essential
```

```
$ gcc --version
```

```
gcc (Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0
```

```
Copyright (C) 2019 Free Software Foundation, Inc.
```

```
This is free software; see the source for copying conditions. There is NO
```

```
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
```

Установка поддержки offload для GCC 9, видеокарта NVidia

```
$ sudo apt-get install gcc-offload-nvptx
```

Установка поддержки offload для GCC 10, видеокарта AMD

```
sudo apt-get install gcc-offload-amdgcn
```

Установка gcc-10, g++10

```
$ sudo apt-get install gcc-10 g++10
```

```
$ gcc-10 --version
```

gcc-10 (Ubuntu 10.2.0-5ubuntu1~20.04) 10.2.0
Copyright (C) 2020 Free Software Foundation, Inc.
This is free software; see the source for copying conditions. There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

\$ g++-10 --version

g++-10 (Ubuntu 10.2.0-5ubuntu1~20.04) 10.2.0

Copyright (C) 2020 Free Software Foundation, Inc.

This is free software; see the source for copying conditions. There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

Установка поддержки offload для GCC 10, видеокарта NVidia

\$ sudo apt-get install gcc-10-offload-nvptx

Установка поддержки offload для GCC 10, видеокарта AMD

sudo apt-get install gcc-10-offload-amdgcn

Установка clang

\$ sudo apt-get install clang

\$ clang --version

clang version 10.0.0-4ubuntu1

Target: x86_64-pc-linux-gnu

Thread model: posix

InstalledDir: /usr/bin

Установка clang-11

\$ sudo apt-get install clang

\$ clang-11 --version

Ubuntu clang version 11.0.0-2~ubuntu20.04.1

Target: x86_64-pc-linux-gnu

Thread model: posix

InstalledDir: /usr/bin

Установка NVIDIA HPC SDK 21.3

Инструкции у установке: <https://developer.nvidia.com/nvidia-hpc-sdk-downloads>

Скачайте два пакета с сайта:

\$ wget https://developer.download.nvidia.com/hpc-sdk/21.3/nvhpc-21-3_21.3_amd64.deb \

https://developer.download.nvidia.com/hpc-sdk/21.3/nvhpc-2021_21.3_amd64.deb

Установите пакеты:

\$ apt-get install ./nvhpc-21-3_21.3_amd64.deb ./nvhpc-2021_21.3_amd64.deb

Настройте переменные окружения согласно руководству

NVIDIA HPC SDK Installation Guide - <https://docs.nvidia.com/hpc-sdk/hpc-sdk-install-guide/index.html#abstract>

\$ NVARCH=`uname -s`_`uname -m`

\$ NVCOMPILERS=/opt/nvidia/hpc_sdk; export NVCOMPILERS

\$ MANPATH=\$MANPATH:\$NVCOMPILERS/\$NVARCH/21.3/compiler/man; export
MANPATH

```
$ PATH=$NVCOMPILERS/$NVARCH/21.3/compilers/bin:$PATH; export PATH
```

Проверка установки переменных окружения:

```
$ echo $NVARCH
Linux_x86_64
$ echo $NVCOMPILERS
/opt/nvidia/hpc_sdk
$ echo $MANPATH
:/opt/nvidia/hpc_sdk/Linux_x86_64/21.3/compilers/man
$ echo $PATH
/opt/nvidia/hpc_sdk/Linux_x86_64/21.3/compilers/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin
```

Проверка версии компилятора (абсолютный путь и с использованием переменной окружения PATH):

```
$ /opt/nvidia/hpc_sdk/Linux_x86_64/21.3/compilers/bin/nvc++ --version
nvc++ 21.3-0 LLVM 64-bit target on x86-64 Linux -tp nehalem
NVIDIA Compilers and Tools
Copyright (c) 2020, NVIDIA CORPORATION. All rights reserved.
$ nvc++ --version
nvc++ 21.3-0 LLVM 64-bit target on x86-64 Linux -tp nehalem
NVIDIA Compilers and Tools
Copyright (c) 2020, NVIDIA CORPORATION. All rights reserved.
```

Установка программного окружения OpenGL и GLUT

```
$ sudo apt-get install freeglut3 freeglut3-dev libglew1.5 libglew1.5-dev libglu1-mesa
libglu1-mesa-dev libgl1-mesa-glx libgl1-mesa-dev
```

Настройка Code::Blocks

Настройка Code::Blocks для использования установленных компиляторов

Компиляторы GCC версии 9 (см. Рис 1), и Clang версии 10 (см. Рис 2), настроены по умолчанию.

GNU GCC 10 Compiler

Создание настроек для GCC 10. Выберите "GNU GCC Compiler", и нажмите кнопку "Сору", в появившемся окне "Add new Compiler", введите название "GNU GCC 10 Compiler", и нажмите "Ок" см. Рис 3. Откройте закладку "Toolchain executables" и установите настройки согласно Рис 4.

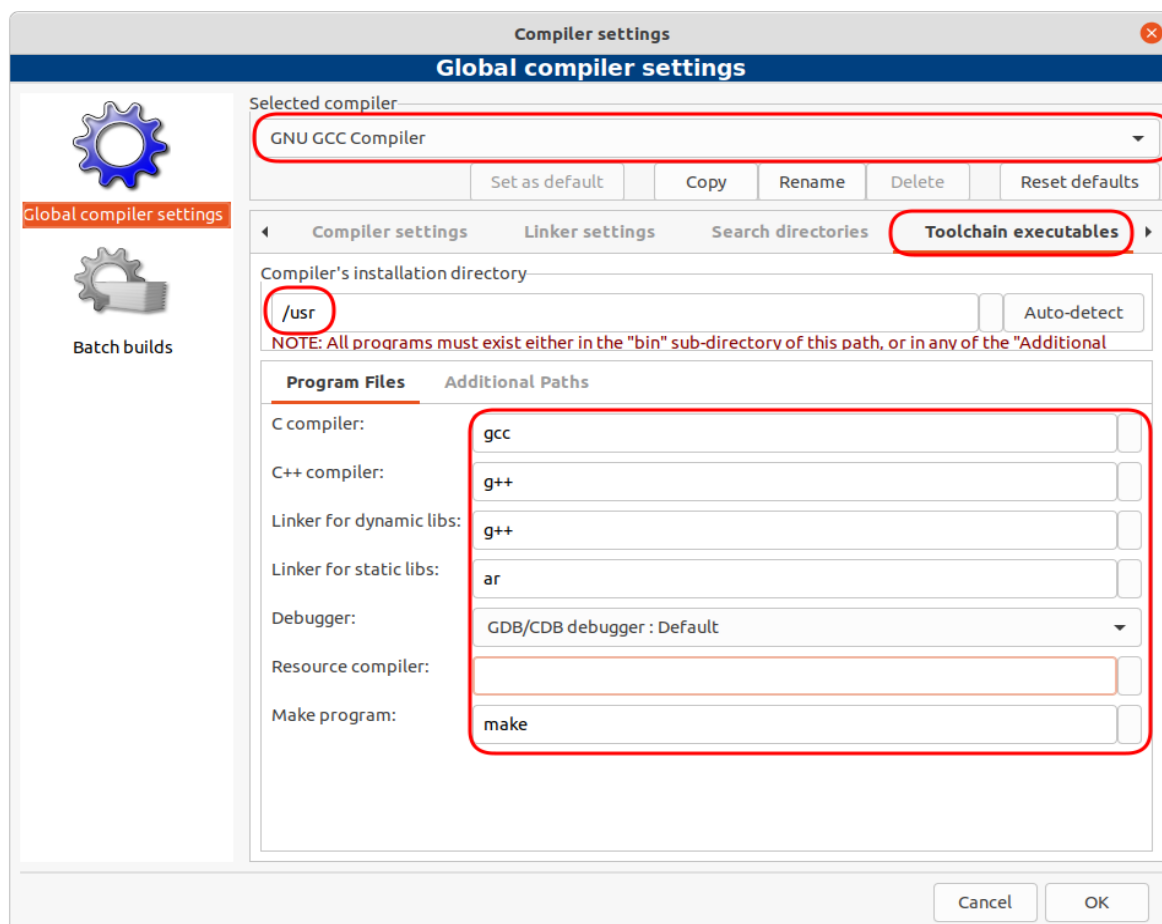


Рис 1 Настройки по умолчанию (GCC — 9)

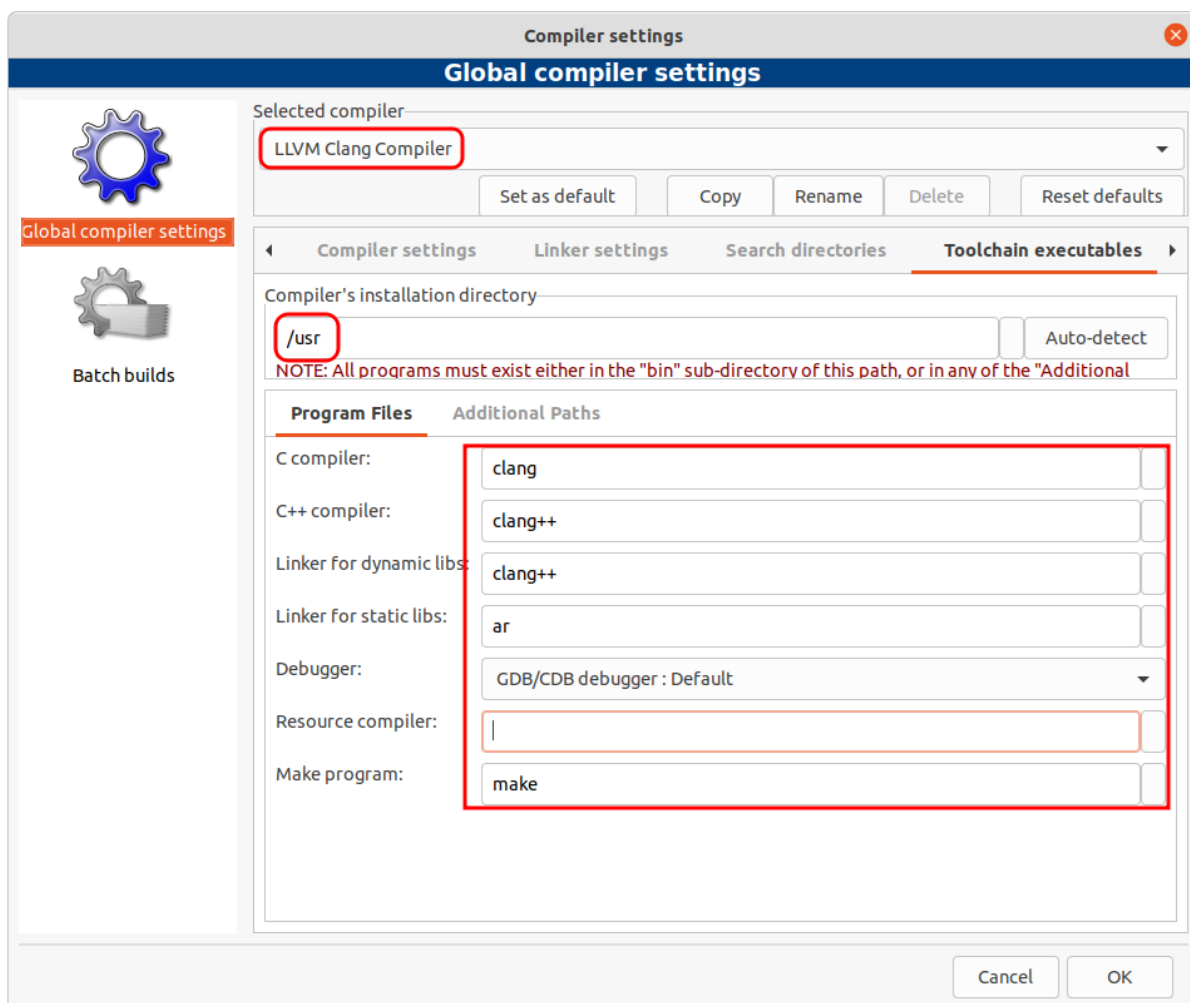


Рис 2 Настройки по умолчанию (Clang 10)

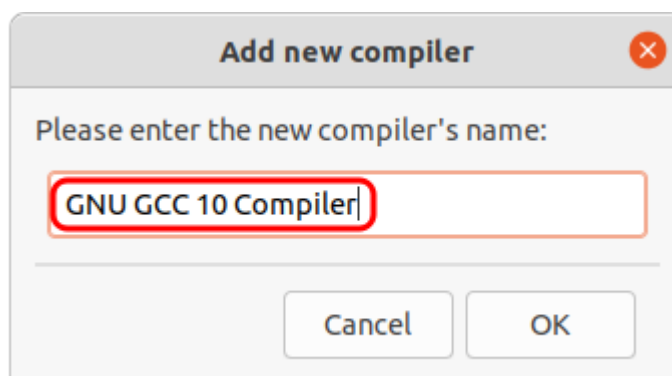


Рис 3: Добавление компилятора GNU GCC 10 Compiler

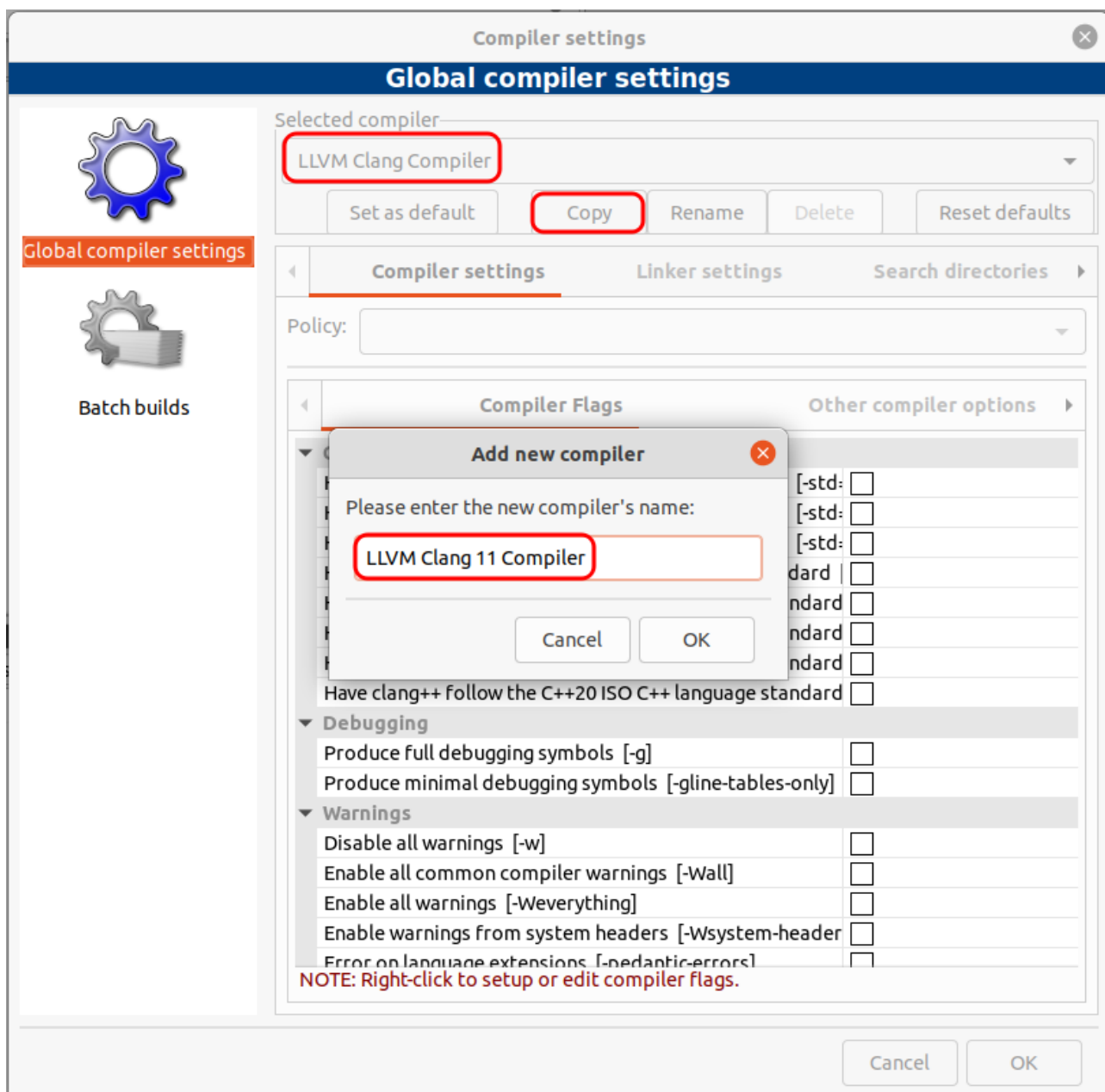


Рис 5. Добавление компилятора LLVM Clang 11 Compiler

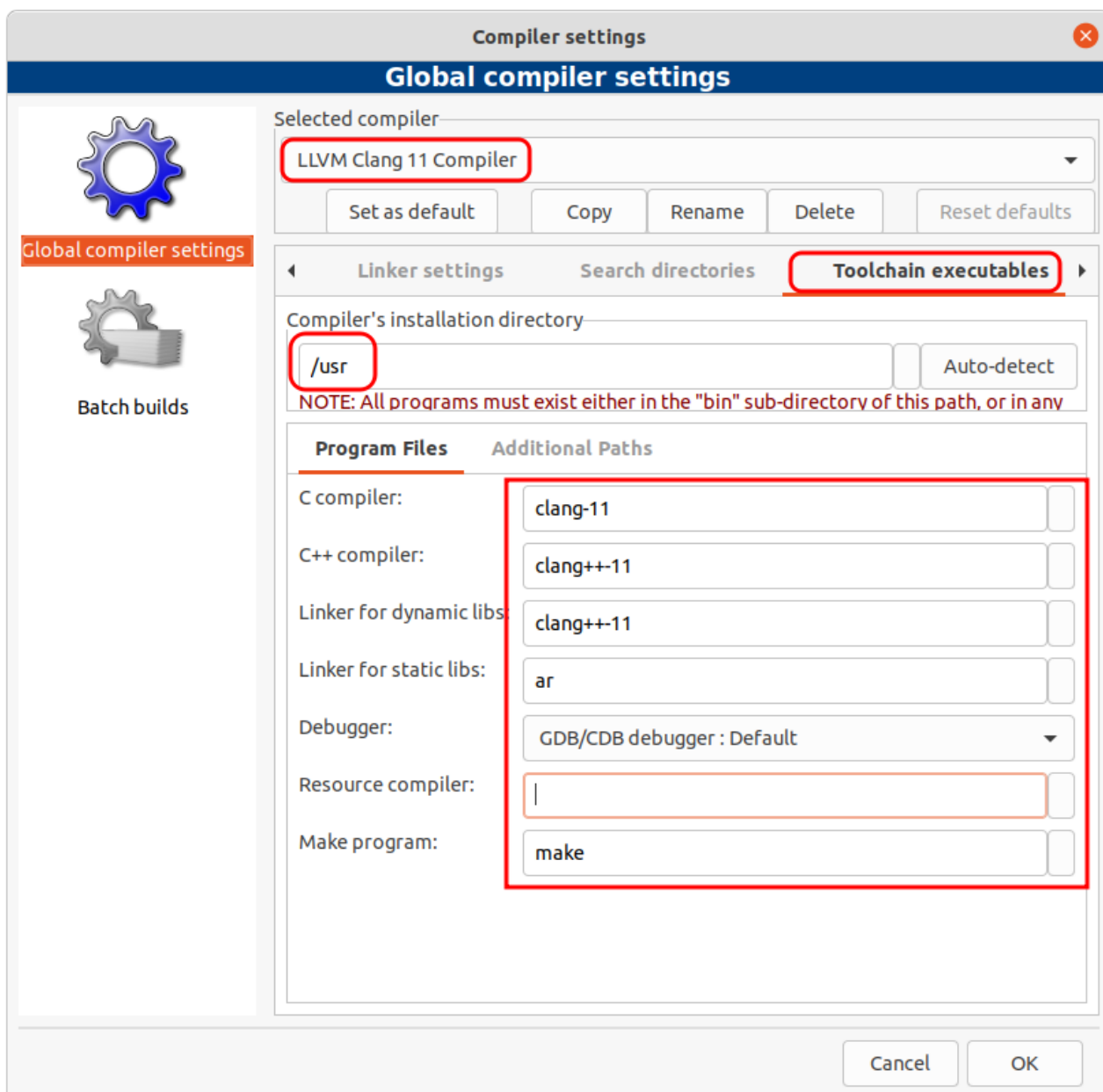


Рис 6: Настройка LLVM Clang 11 Compiler

Nvidia nvcc++ 21.3-0 LLVM

Создание настроек для Nvidia nvcc++ 21.3-0 LLVM. Выберите "LLVM Clang Compiler", и нажмите кнопку "Copy", в появившемся окне "Add new Compiler", введите название "Nvidia nvcc++ 21.3-0 LLVM", и нажмите "Ок" см Рис 7. Откройте закладку "Toolchain executables" и установите настройки согласно см. Рис 8.

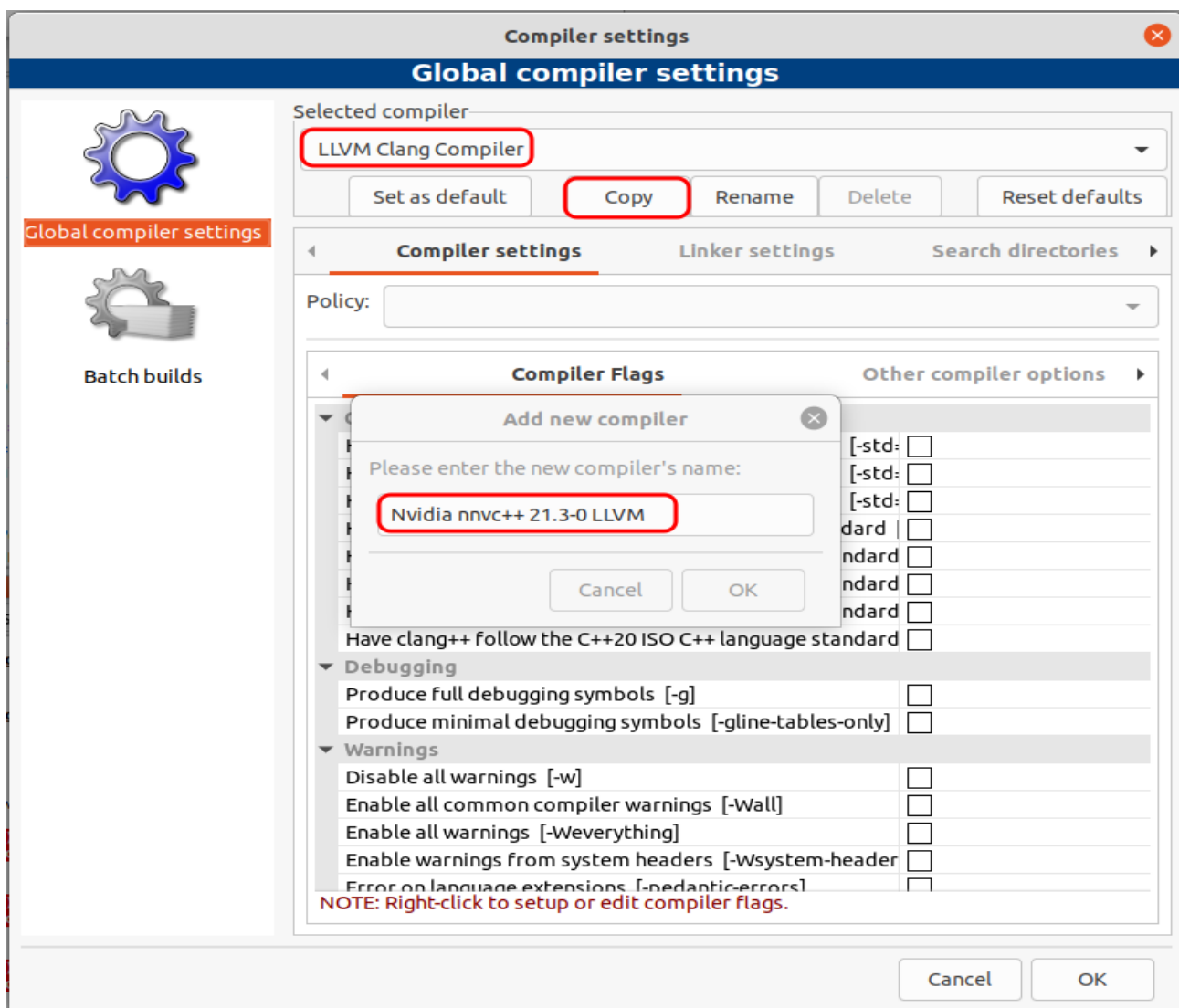


Рис 7: Добавление компилятора Nvidia nvcc++ 21.3-0 LLVM

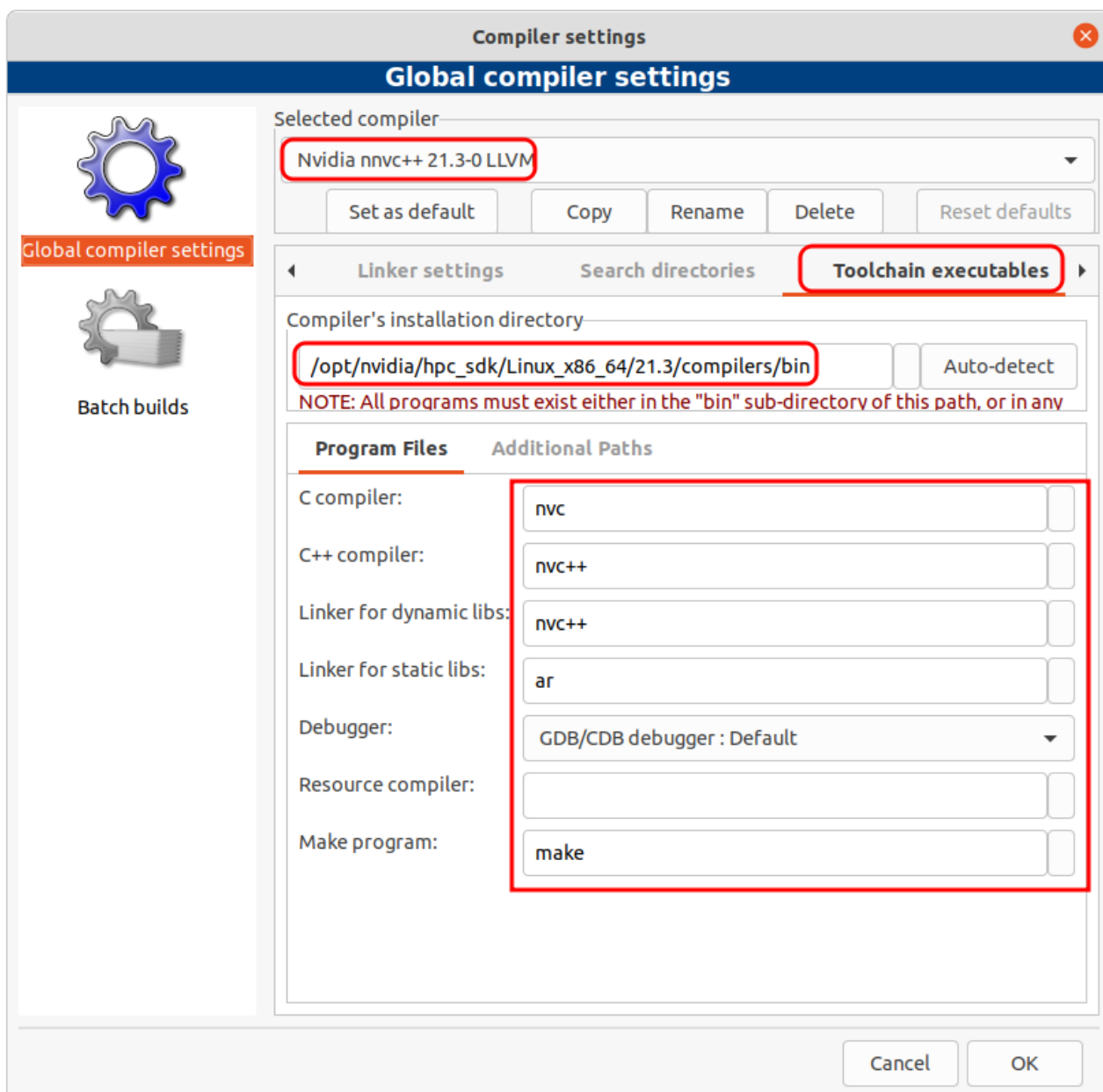


Рис 8 Настройки Nvidia nvcc++ 21.3-0 LLVM

Настройка опций сборки в Code::Blocks для OpenMP и OpenACC с использованием установленных компиляторов

Компиляторы GCC и NVCC++ Clang поддерживают OpenMP, но OpenACC поддерживают только GCC и NVCC++. В нашем случае настройки GCC 9 и GCC 10 не различаются, тоже относится к Clang 10 и Clang 11. В секции опций компилятора "Nvidia nvcc++ 21.3-0 LLVM" необходимо быть осторожным, сверять реально существующие настройки с устанавливаемыми, так как присутствуют настройки специфичные для Clang (были скопированы из настроек данного компилятора).

GCC 9 и GCC 10

OpenACC

Для обеспечения поддержки OpenACC, необходимо добавить опции компилятора в закладке "*Other compiler options*" (пример см. Рис 9):

-ffast-math

-fopenacc

-foffload=nvptx-none

-fcf-protection=none

-foffload="-O3"

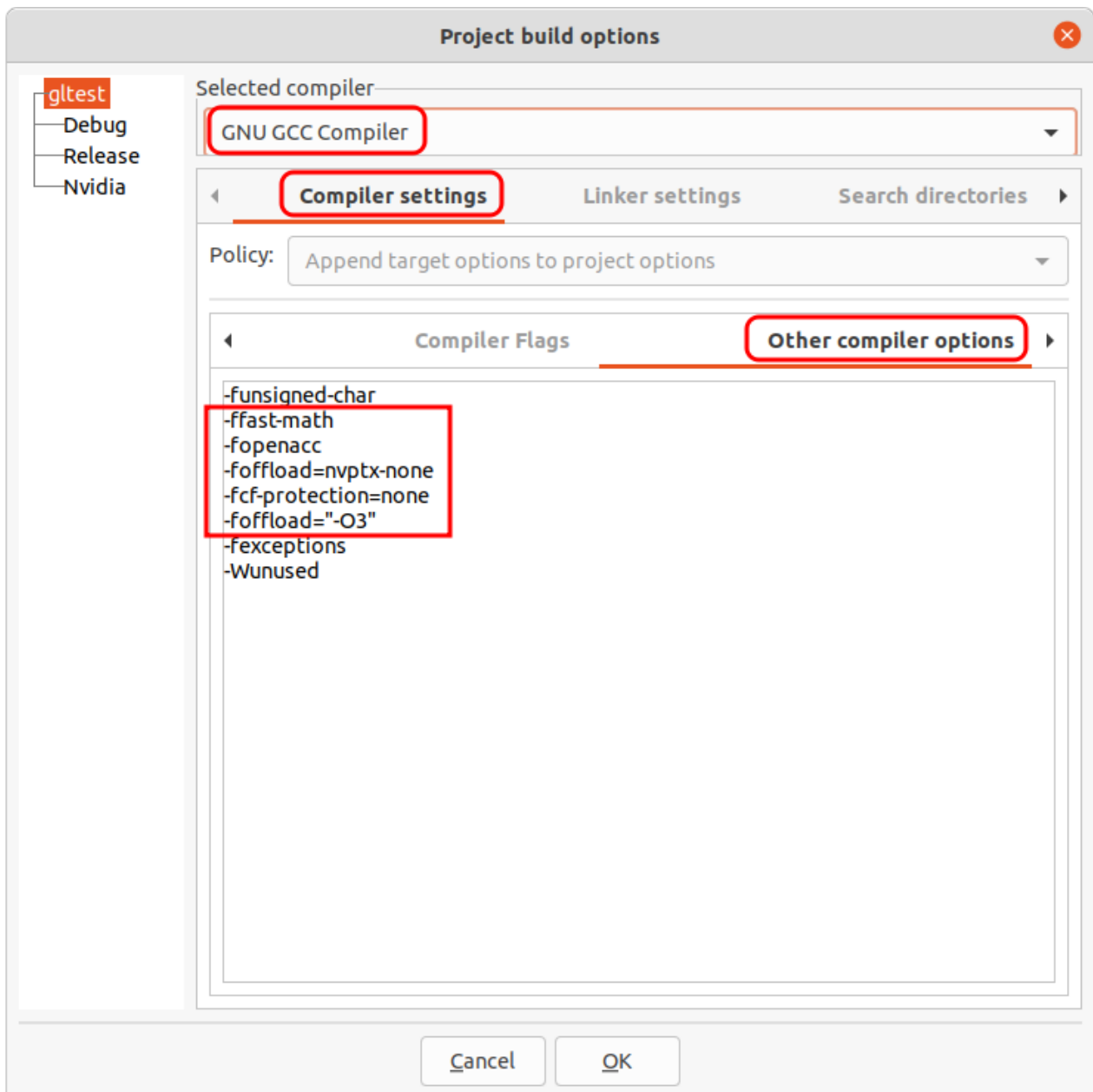


Рис 9: GCC - "Other compiler options" OpenACC

Так же в закладке "Linker settings" и группе "Other linker options" указать нижеприведенные опции (пример см. Рис 10):

- fopenacc
- foffload=nvptx-none
- fcf-protection=none
- foffload="-O3"

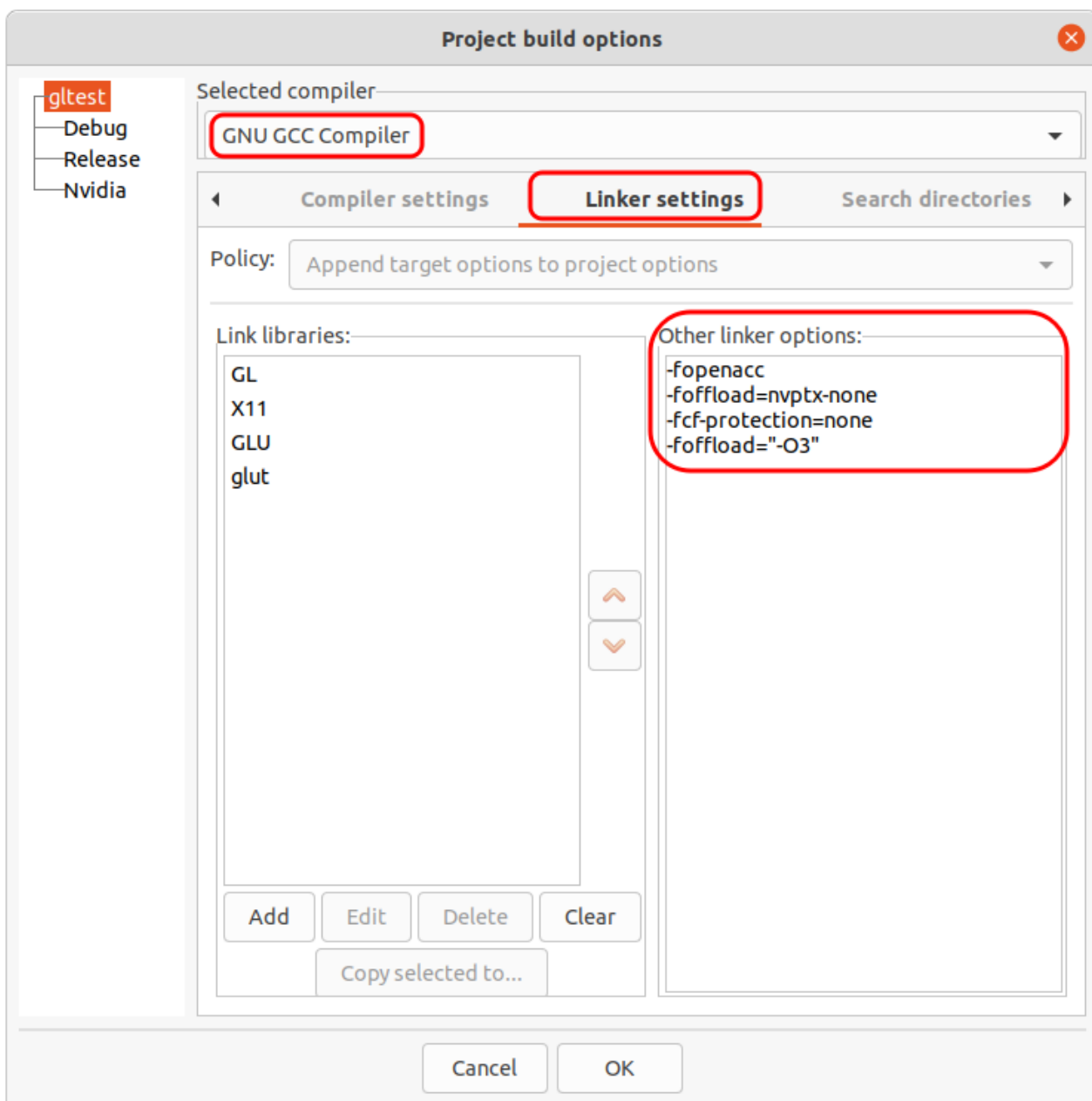


Рис 10: GCC Опции в "Other linker options" OpenACC

OpenMP

Для обеспечения поддержки OpenMP, необходимо добавить опции компилятора в закладке "Compiler settings" и группе "Other compiler options" (пример см. Рис 11):

-fopenmp

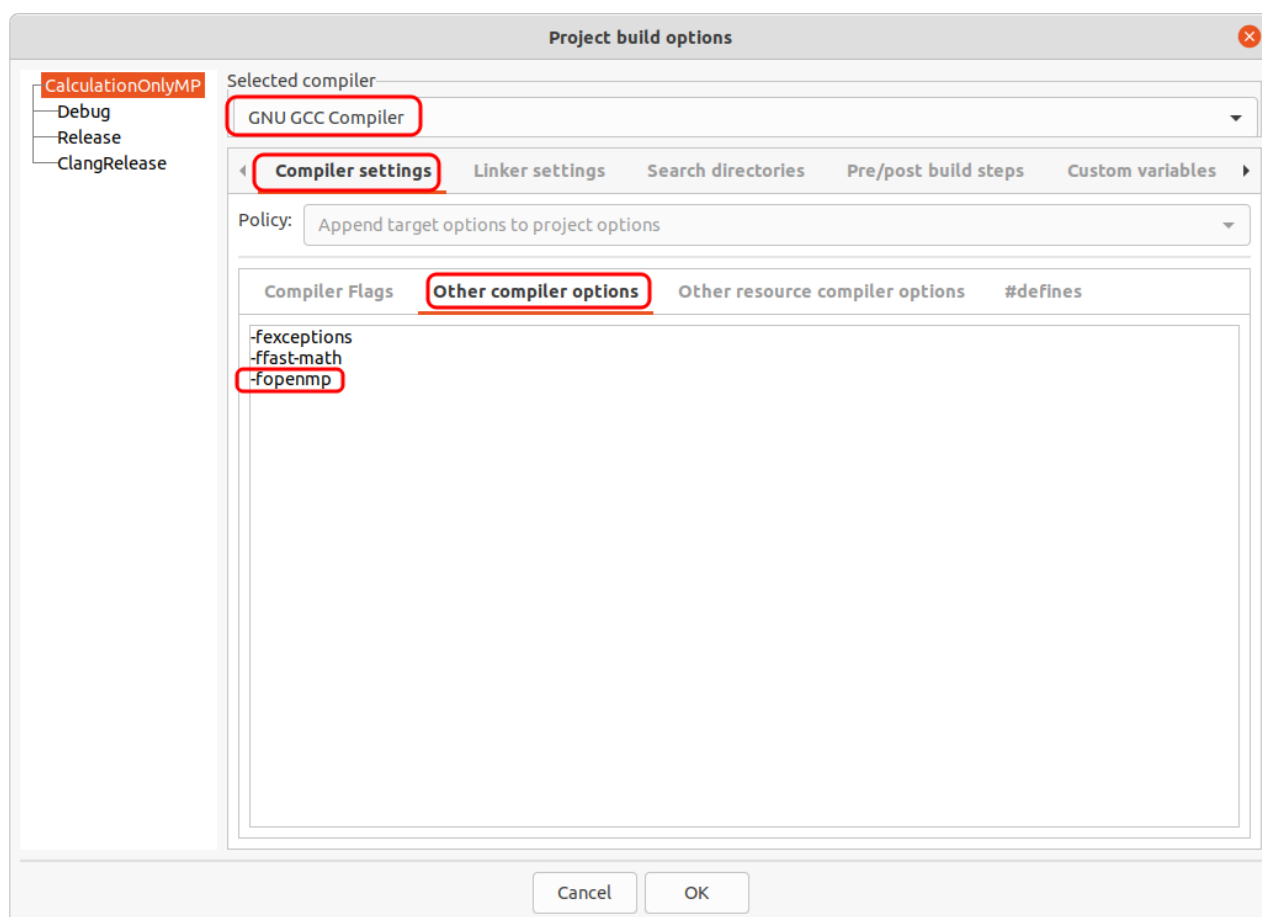


Рис 11: GCC - "Other compiler options" OpenMP

Так же в закладке "*Linker settings*" и группе "*Other linker options*" указать нижеприведенную библиотеку (пример см. Рис 12):

-gomp

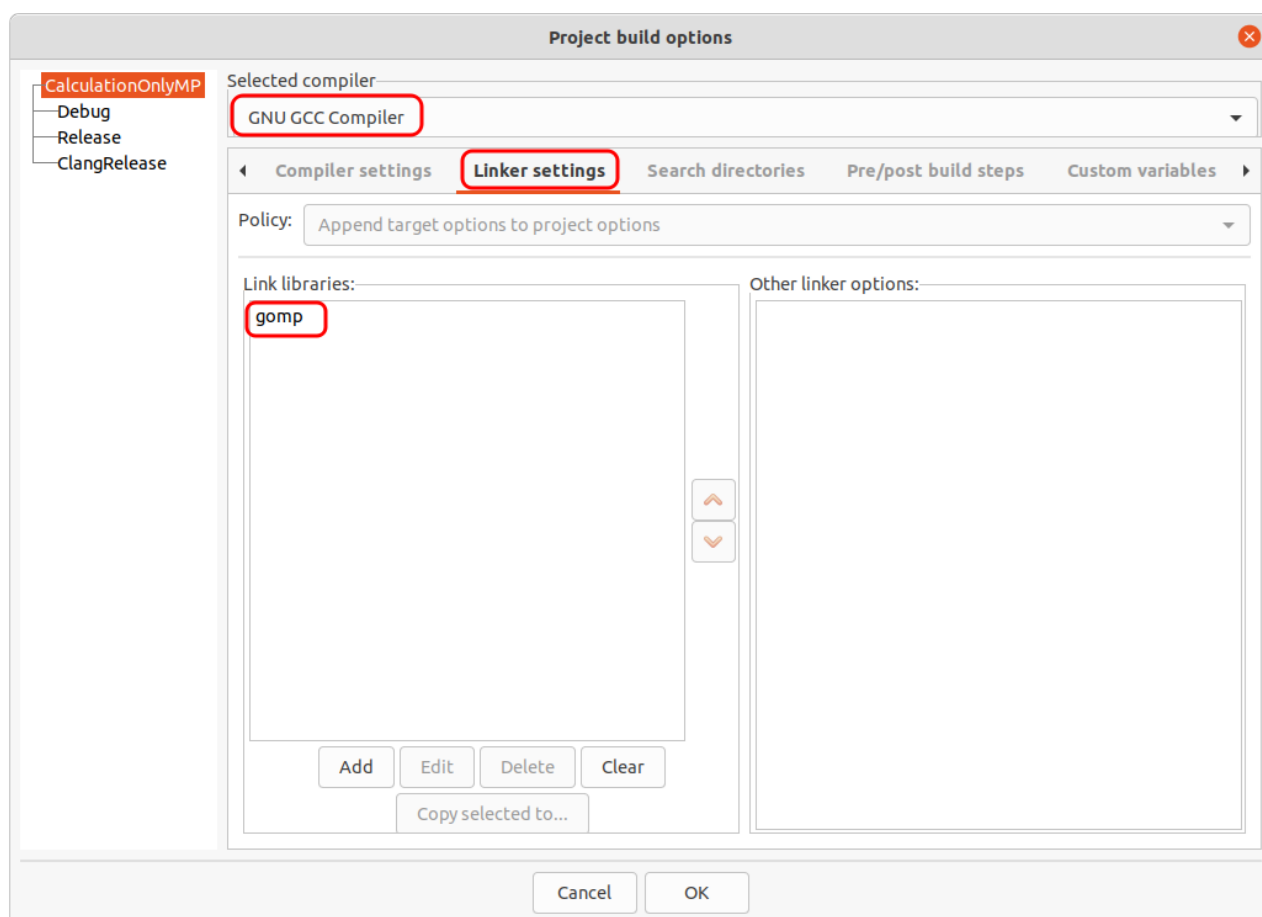


Рис 12: GCC Опции в "Link Libraries" OpenACC

Clang 10 и Clang 11

OpenACC — не поддерживается данными компиляторами.

Для обеспечения поддержки OpenMP, необходимо добавить опцию компилятора в закладке "Compiler settings" и группе "Other compiler options" (пример см. Рис 13):

`-fopenmp=libiomp5`

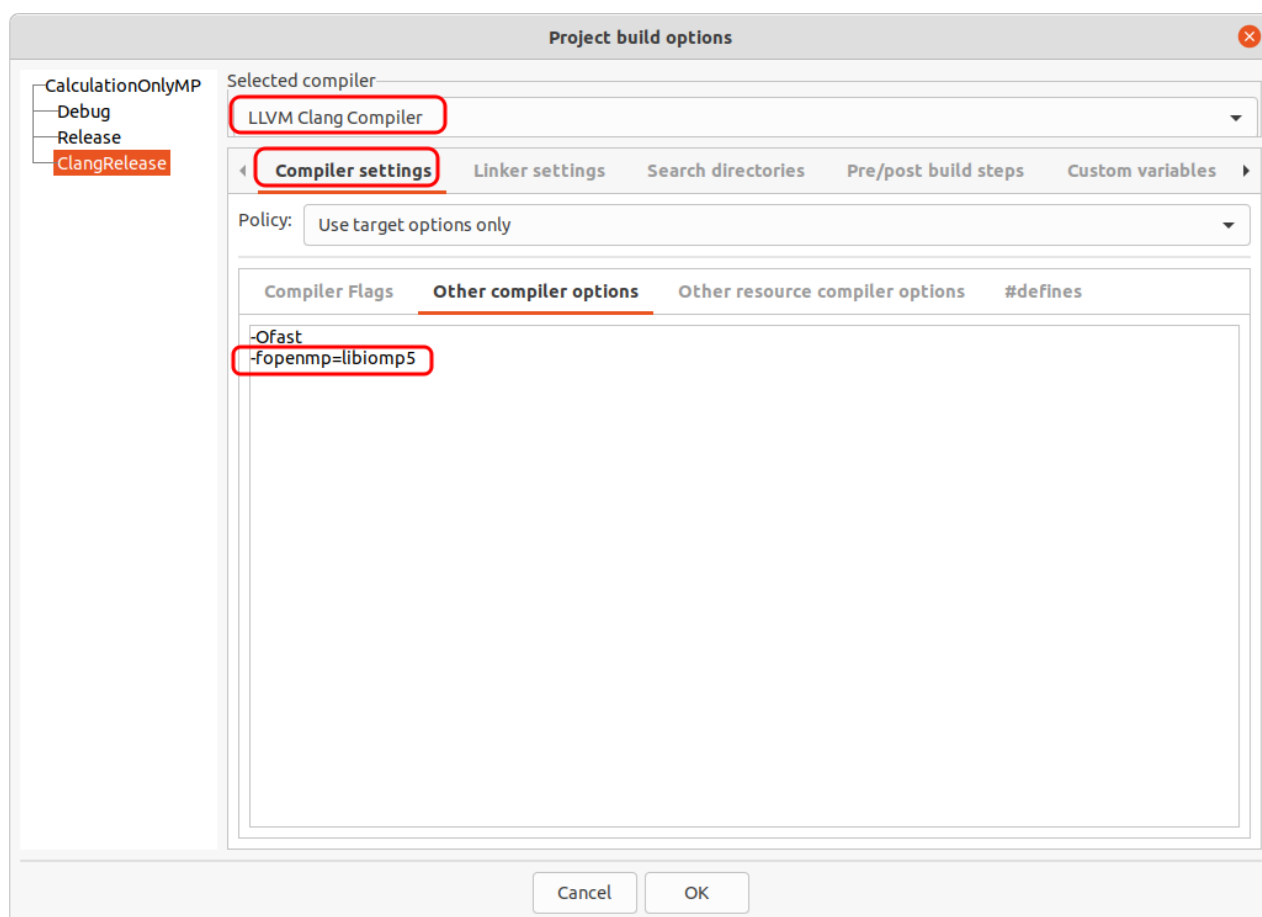


Рис 13. Clang - "Other compiler options" OpenMP

Так же в закладке "*Linker settings*" и группе "*Other linker options*" указать нижеприведенную библиотеку (пример см. Рис 14):

`-fopenmp=libiomp5`

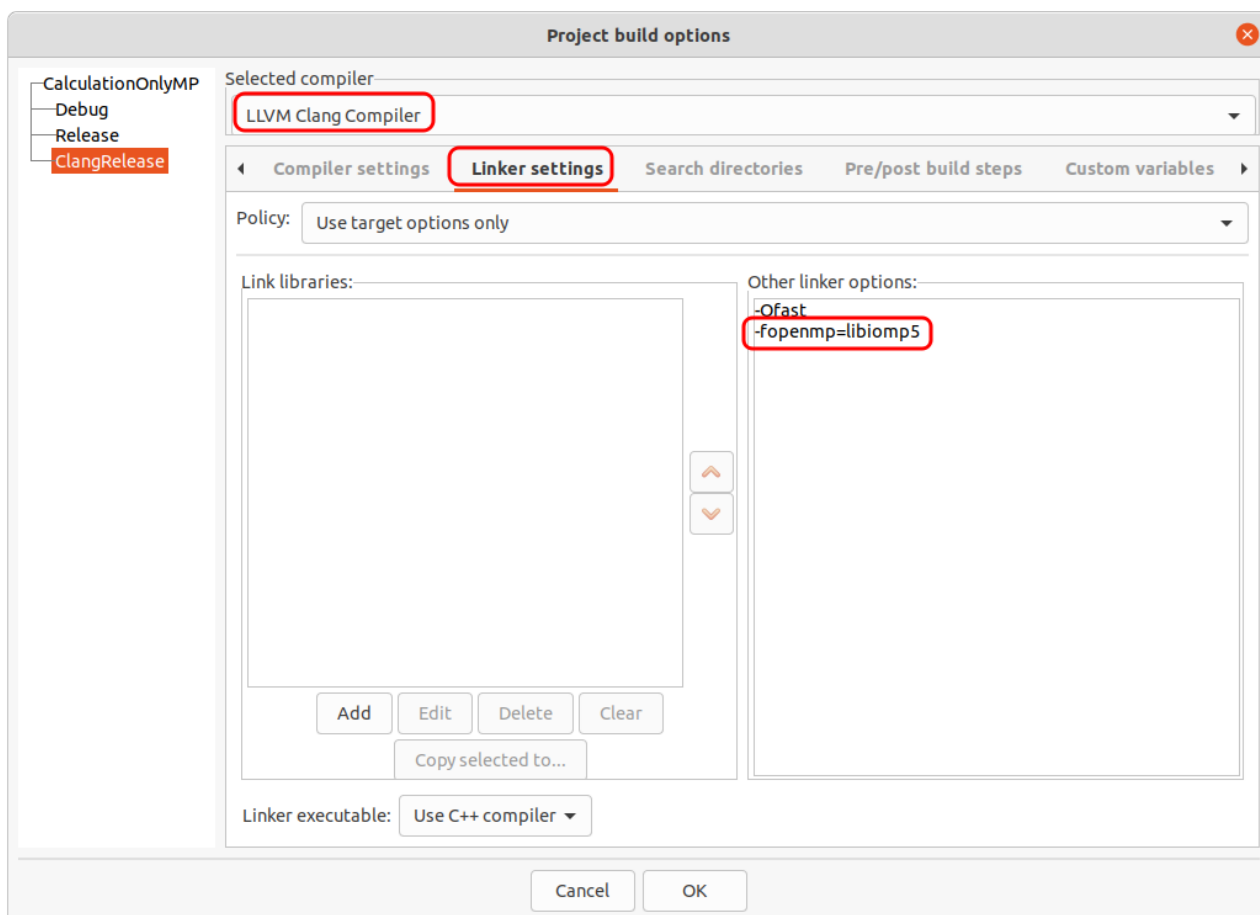


Рис 14: Clang Опции в "Other linker options" OpenMP

NVIDIA HPC Compiler nvc++ 21.3-0 LLVM

OpenACC

Для обеспечения поддержки OpenACC, необходимо добавить опции компилятора в закладке "Other compiler options" (пример см. Рис 15):

-acc

-gpu=cc60

Для **-gpu** следует указать параметры, соответствующие вашей видео-карте, настройки см. <https://docs.nvidia.com/hpc-sdk/compiler/hpc-compilers-user-guide/index.html#gpu>

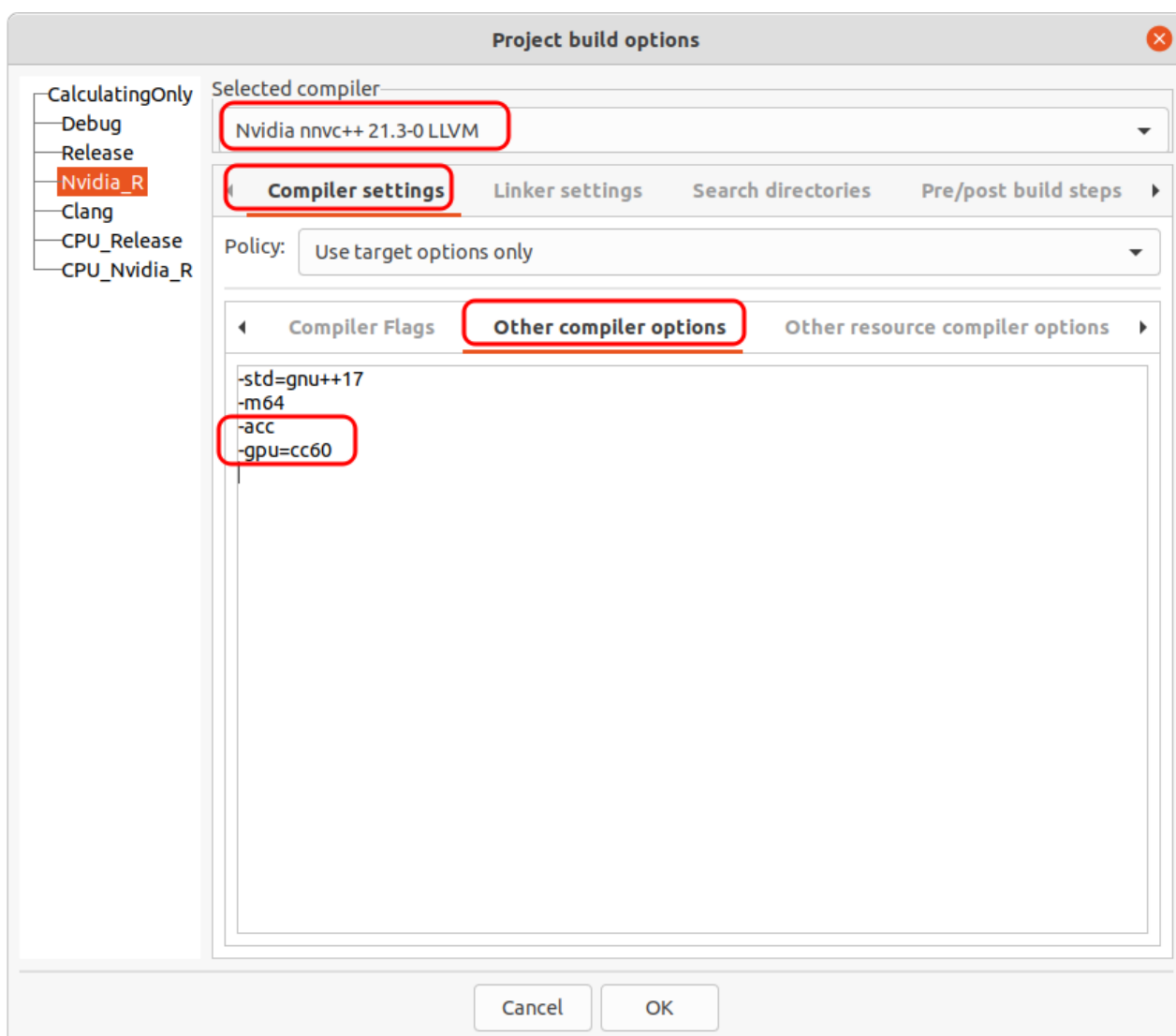


Рис 15: NVIDIA HPC Compiler nvc++ 21.3-0 LLVM - "Other compiler options" OpenACC

Так же в закладке "*Linker settings*" и группе "*Other linker options*" указать нижеприведенные опции (пример см. Рис 16):

-acc

-gpu=cc60

Для **-gpu** следует указать параметры, соответствующие вашей видео-карте, настройки см. <https://docs.nvidia.com/hpc-sdk/compilers/hpc-compilers-user-guide/index.html#gpu>

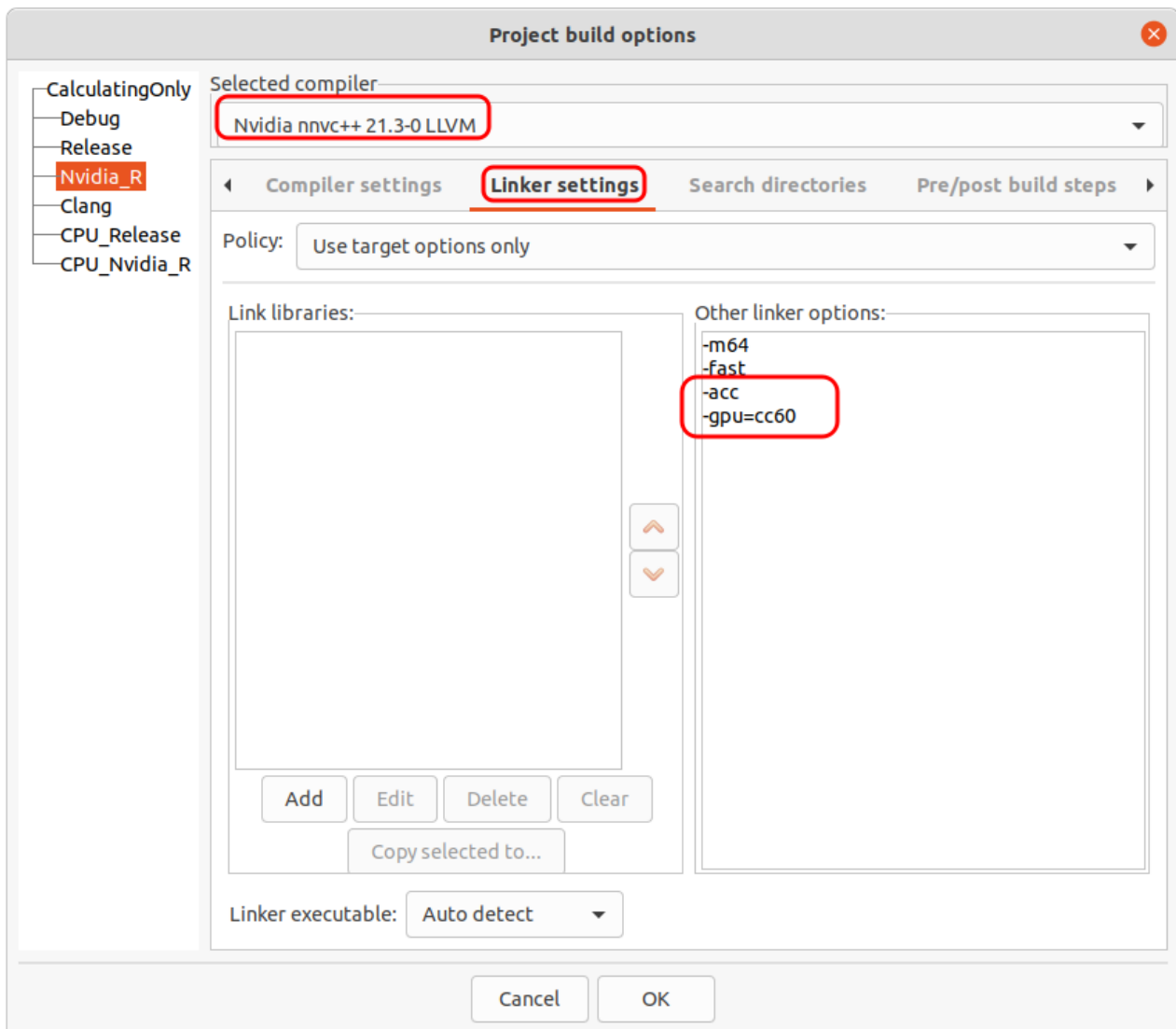


Рис. 16. NVIDIA HPC Compiler nvcc++ 21.3-0 LLVM Опции в "Link Libraries" OpenACC