

C++ Development

Pieter P

To build a C++ application for the Raspberry Pi, you have to follow three main steps: 1. Build a cross-compilation toolchain, 2. Cross-compile the libraries you want to use, 3. Build your actual C++ application.

These pages explain how to build a cross-compilation toolchain using Crosstool-NG, and then uses it to compile the following libraries:

- **Zlib**: compression library (OpenSSL and Python dependency)
- **OpenSSL**: cryptography library (Python dependency)
- **FFI**: foreign function interface (Python dependency, used to call C functions using ctypes)
- **Bzip2**: compression library (Python dependency)
- **GNU ncurses**: library for text-based user interfaces (Python dependency, used for the console)
- **GNU readline**: library for line-editing and history (Python dependency, used for the console)
- **GNU dbm**: library for key-value data (Python dependency)
- **SQLite**: library for embedded databases (Python dependency)
- **UUID**: library for unique identifiers (Python dependency)
- **Python 3.8.1**: Python interpreter and libraries
- **ZBar**: Bar and QR code decoding library
- **Raspberry Pi Userland**: VideoCore GPU drivers
- **VPX**: VP8/VP9 codec SDK
- **x264**: H.264/MPEG-4 AVC encoder
- **Xvid**: MPEG-4 video codec
- **FFmpeg**: library to record, convert and stream audio and video
- **OpenBLAS**: linear algebra library (NumPy dependency)
- **NumPy**: multi-dimensional array container for Python (OpenCV dependency)
- **SciPy**: Python module for mathematics, science, and engineering
- **OpenCV 4.2.0**: computer vision library and Python module
- **GDB Server**: on-target remote debugger
- **GCC 9.2.0**: C, C++ and Fortran compilers
- **GNU Make**: build automation tool
- **CMake**: build system
- **Distcc**: distributed compiler wrapper (uses your computer to speed up compilation on the RPi)
- **CCache**: compiler cache
- **cURL**: tool and library for transferring data over the network (Git dependency)
- **Git**: version control system

The fourth page presents a small "hello world" example project that uses CMake and Google Test.

Finally, there's a page on remote on-target debugging using GDB and Visual Studio Code.

Building the Cross-Compilation Toolchain

To compile software for the Raspberry Pi, you need a cross-compilation toolchain. This page contains instructions for how to build one.

Cross-Compiling the Dependencies

Using the cross-compilation toolchain to build the libraries you need for your project, as well as their dependencies.

Setting up Visual Studio Code for C++ Development

Installing and configuring the right extensions for easy C++ development in VSCode.

Cross-Compiling the C++ Example Project

Using the cross-compilation toolchain to build your own C++ project.
