Usage

To build dependencies for the current arch+OS:

```
make
```

To build for another arch/OS:

```
make HOST=host-platform-triplet
```

For example:

```
make HOST=x86_64-w64-mingw32 -j4
```

Bitcoin Core's `configure` script by default will ignore the depends output. In order for it to pick up libraries, tools, and settings from the depends build, you must set the `CONFIG_SITE` environment variable to point to a `config.site` settings file. In the above example, a file named `depends/x86_64-w64-mingw32/share/config.site` will be created. To use it during compilation:

```
CONFIG_SITE=$PWD/depends/x86_64-w64-mingw32/share/config.site ./configure
```

The default install prefix when using `config.site` is `--prefix=depends/<host-platform-triplet>`, so depends build outputs will be installed in that location.

Common `host-platform-triplet`s for cross compilation are:

- `i686-pc-linux-gnu` for Linux 32 bit
- `x86_64-pc-linux-gnu` for x86 Linux
- `x86_64-w64-mingw32` for Win64
- `x86_64-apple-darwin` for macOS
- `arm64-apple-darwin` for ARM macOS
- `arm-linux-gnueabihf` for Linux ARM 32 bit
- `aarch64-linux-gnu` for Linux ARM 64 bit
- `powerpc64-linux-gnu` for Linux POWER 64-bit (big endian)
- `powerpc64le-linux-gnu` for Linux POWER 64-bit (little endian)
- `riscv32-linux-gnu` for Linux RISC-V 32 bit
- `riscv64-linux-gnu` for Linux RISC-V 64 bit
- `s390x-linux-gnu` for Linux S390X
- `armv7a-linux-android` for Android ARM 32 bit
- `aarch64-linux-android` for Android ARM 64 bit
- `x86_64-linux-android` for Android x86 64 bit

The paths are automatically configured and no other options are needed unless targeting Android.

Install the required dependencies: Ubuntu & Debian

For macOS cross compilation

```
sudo apt-get install curl bsdmainutils cmake libz-dev python3-setuptools libtinfo5 xorriso
```

Note: You must obtain the macOS SDK before proceeding with a cross-compile. Under the depends directory, create a subdirectory named `SDKs`. Then, place the extracted SDK under this new directory. For more information, see SDK Extraction.

For Win64 cross compilation

- see build-windows.md

For linux (including i386, ARM) cross compilation Common linux dependencies:

```
sudo apt-get install make automake cmake curl g++-multilib libtool binutils-gold bsdmainuti
```

For linux ARM cross compilation:

```
sudo apt-get install g++-arm-linux-gnueabihf binutils-arm-linux-gnueabihf
```

For linux AARCH64 cross compilation:

```
sudo apt-get install g++-aarch64-linux-gnu binutils-aarch64-linux-gnu
```

For linux POWER 64-bit cross compilation (there are no packages for 32-bit):

```
sudo apt-get install g++-powerpc64-linux-gnu binutils-powerpc64-linux-gnu g++-powerpc64le-li
```

For linux RISC-V 64-bit cross compilation (there are no packages for 32-bit):

```
sudo apt-get install g++-riscv64-linux-gnu binutils-riscv64-linux-gnu
```

For linux S390X cross compilation:

```
sudo apt-get install g++-s390x-linux-gnu binutils-s390x-linux-gnu
```

Install the required dependencies: OpenBSD

```
pkg_add bash gtar
```

Dependency Options

The following can be set when running make: `make FOO=bar`

- `SOURCES_PATH`: Downloaded sources will be placed here
- `BASE_CACHE`: Built packages will be placed here
- `SDK_PATH`: Path where SDKs can be found (used by macOS)
- `FALLBACK_DOWNLOAD_PATH`: If a source file can't be fetched, try here before giving up
- `NO_QT`: Don't download/build/cache Qt and its dependencies
- `NO_QR`: Don't download/build/cache packages needed for enabling qrencode
- `NO_ZMQ`: Don't download/build/cache packages needed for enabling ZeroMQ
- `NO_WALLET`: Don't download/build/cache libs needed to enable the wallet
- `NO_BDB`: Don't download/build/cache BerkeleyDB

- `NO_SQLITE`: Don't download/build/cache SQLite
- `NO_UPNP`: Don't download/build/cache packages needed for enabling UPnP
- `NO_NATPMP`: Don't download/build/cache packages needed for enabling NAT-PMP
- `ALLOW_HOST_PACKAGES`: Packages that are missed in dependencies (due to `NO_*` option or build script logic) are searched for among the host system packages using `pkg-config`. It allows building with packages of other (newer) versions
- `MULTIPROCESS`: Build libmultiprocess (experimental, requires CMake)
- `DEBUG`: Disable some optimizations and enable more runtime checking
- `HOST_ID_SALT`: Optional salt to use when generating host package ids
- `BUILD_ID_SALT`: Optional salt to use when generating build package ids
- `FORCE_USE_SYSTEM_CLANG`: (EXPERTS ONLY) When cross-compiling for macOS, use Clang found in the system's `$PATH` rather than the default prebuilt release of Clang from llvm.org. Clang 8 or later is required.

If some packages are not built, for example `make NO_WALLET=1`, the appropriate options will be passed to bitcoin's configure. In this case, `--disable-wallet`.

Additional targets

```
download: run 'make download' to fetch all sources without building them
download-osx: run 'make download-osx' to fetch all sources needed for macOS builds
download-win: run 'make download-win' to fetch all sources needed for win builds
download-linux: run 'make download-linux' to fetch all sources needed for linux builds
```

Other documentation

- description.md: General description of the depends system
- packages.md: Steps for adding packages