

Yocto recipe

```
inbeledl2609cs @ meta-crow-asio
```

```
$ tree
```

```
.
├── conf
│   └── layer.conf
├── COPYING.MIT
├── README
├── recipes-crow-asio
│   └── crow-asio
│       └── crow-asio_1.0.0.bb
└── recipes-example
    ├── crow-test
    │   ├── crow-test_1.0.bb
    │   └── files
    │       ├── CMakeLists.txt
    │       └── main.cpp
    └── example
        └── example_0.1.bb
```

8 directories, 8 files

```
crow-asio_1.0.0.bb
```

```
SUMMARY = "Crow C++ web framework with ASIO networking library"
```

```
DESCRIPTION = "A fast and easy to use C++ micro web framework with  
standalone ASIO networking library"
```

```
HOMEPAGE = "https://github.com/CrowCpp/Crow"
```

```
SECTION = "libs"
```

```
LICENSE = "BSD-3-Clause"
```

```
LIC_FILES_CHKSUM = "file://${COMMON_LICENSE_DIR}/BSD-3-  
Clause;md5=550794465ba0ec5312d6919e203a55f9"
```

```
SRC_URI = "https://github.com/chriskohlhoff/asio/archive/refs/tags/asio-1-  
30-2.tar.gz;name=asio \  
           https://github.com/CrowCpp/Crow/releases/download/v1.2.1.2/Crow-  
1.2.1-Linux.tar.gz;name=crow"
```

```
SRC_URI[asio.sha256sum] =
```

```
"755bd7f85a4b269c67ae0ea254907c078d408cce8e1a352ad2ed664d233780e8"
```

```
SRC_URI[crow.sha256sum] =
```

```
"242d72a809f48bb3169c1361798e81e3f553117cced107d4a309748672cebf23"
```

```
S = "${WORKDIR}"
```

```
DEPENDS = "cmake-native"
```

```
do_unpack:append() {  
    # Reorganize extracted files to match expected structure  
    import os, shutil  
    workdir = d.getVar('WORKDIR')  
  
    # Create organized directory structure  
    asio_dir = os.path.join(workdir, 'asio')  
    crow_dir = os.path.join(workdir, 'crow')  
  
    # Move ASIO to expected location  
    if os.path.exists(os.path.join(workdir, 'asio-asio-1-30-2')):  
        if not os.path.exists(asio_dir):  
            shutil.move(os.path.join(workdir, 'asio-asio-1-30-2'), asio_dir)  
  
    # Move Crow to expected location  
    if os.path.exists(os.path.join(workdir, 'Crow-1.2.1-Linux')):  
        if not os.path.exists(crow_dir):  
            shutil.move(os.path.join(workdir, 'Crow-1.2.1-Linux'), crow_dir)  
}
```

```
do_install() {  
    # Create include directory for header-only libraries  
    install -d ${D}${includedir}  
  
    # Install ASIO headers  
    install -d ${D}${includedir}/asio  
    cp -r ${WORKDIR}/asio/asio/include/* ${D}${includedir}/asio/  
  
    # Install Crow headers  
    install -d ${D}${includedir}/crow  
    cp -r ${WORKDIR}/crow/include/* ${D}${includedir}/crow/  
  
    # Create pkg-config files for easier integration  
    install -d ${D}${libdir}/pkgconfig
```

```

    # Create asio.pc
    cat > ${D}${libdir}/pkgconfig/asio.pc << EOF
prefix=${prefix}
exec_prefix=\${prefix}
includedir=${includedir}

Name: asio
Description: Asio C++ Library
Version: 1.30.2
Cflags: -I\${includedir}/asio
EOF

    # Create crow.pc
    cat > ${D}${libdir}/pkgconfig/crow.pc << EOF
prefix=${prefix}
exec_prefix=\${prefix}
includedir=${includedir}

Name: crow
Description: Crow C++ micro web framework
Version: 1.2.1.2
Requires: asio
Cflags: -I\${includedir}/crow
EOF
}

FILES_${PN} = "${includedir}/asio/* ${includedir}/crow/*
${libdir}/pkgconfig/*"
FILES_${PN}-dev = "${includedir}/asio/* ${includedir}/crow/*
${libdir}/pkgconfig/*"

# These are header-only libraries
RDEPENDS_${PN}-dev = ""
RRECOMMENDS_${PN}-dbg = ""

BBCLASSEXTEND = "native nativesdk"

```

```
crow-test_1.0.bb
```

```

SUMMARY = "Simple Crow C++ web server test application"
DESCRIPTION = "A test application demonstrating the use of Crow C++
framework with ASIO"
HOMEPAGE = "https://github.com/CrowCpp/Crow"

```

```

SECTION = "examples"

LICENSE = "MIT"
LIC_FILES_CHKSUM =
"file://${COMMON_LICENSE_DIR}/MIT;md5=0835ade698e0bcf8506ecda2f7b4f302"

DEPENDS = "crow-asio pkgconfig-native openssl zlib"

SRC_URI = "file://main.cpp \
           file://CMakeLists.txt"

S = "${WORKDIR}"

inherit cmake

do_install() {
    install -d ${D}${bindir}
    install -m 0755 ${B}/crow-test ${D}${bindir}/
}

FILES_${PN} = "${bindir}/crow-test"

```

CMakeLists.txt

```

cmake_minimum_required(VERSION 3.16)
project(crow-test)

set(CMAKE_CXX_STANDARD 17)
set(CMAKE_CXX_STANDARD_REQUIRED ON)

# Find required packages
find_package(PkgConfig REQUIRED)
pkg_check_modules(ASIO REQUIRED asio)
pkg_check_modules(CROW REQUIRED crow)
find_package(OpenSSL REQUIRED)
find_package(ZLIB REQUIRED)

# Add executable
add_executable(crow-test main.cpp)

# Set include directories
target_include_directories(crow-test PRIVATE
    ${ASIO_INCLUDE_DIRS}

```

```
    ${CROW_INCLUDE_DIRS}
)

# Add compile definitions
target_compile_definitions(crow-test PRIVATE
    CROW_ENABLE_SSL=1
    ASIO_STANDALONE
)

# Link libraries
find_package(Threads REQUIRED)
target_link_libraries(crow-test
    Threads::Threads
    OpenSSL::SSL
    OpenSSL::Crypto
    ZLIB::ZLIB
)
```

main.cpp

```
#include "crow.h"

int main()
{
    crow::SimpleApp app;

    CROW_ROUTE(app, "/")([](){
        return "Hello World! Crow C++ is working with ASIO!";
    });

    CROW_ROUTE(app, "/json")([]{
        crow::json::wvalue x;
        x["message"] = "Hello from Crow with ASIO!";
        x["framework"] = "Crow C++";
        x["networking"] = "ASIO";
        return x;
    });

    CROW_ROUTE(app, "/health")([]{
        return "OK";
    });

    app.port(8080).multithreaded().run();
}
```

```
    return 0;
}
```

meta-crow-asio Layer

This layer provides Crow C++ web framework and ASIO networking library for Yocto builds.

Overview

The meta-crow-asio layer includes:

- **ASIO**: Asynchronous I/O C++ library (version 1.30.2)
- **Crow**: Fast and easy to use C++ micro web framework (version 1.2.1.2)
- Test applications demonstrating usage

Dependencies

This layer depends on:

- meta-openembedded (for cmake support)
- openembedded-core

Recipes Provided

crow-asio (main recipe)

- **Recipe**: `recipes-crow-asio/crow-asio/crow-asio_1.0.0.bb`
- **Description**: Downloads and installs Crow C++ framework with ASIO networking library
- **Sources**:
 - ASIO: <https://github.com/chriskohlhoff/asio/archive/refs/tags/asio-1-30-2.tar.gz>
 - Crow: <https://github.com/CrowCpp/Crow/releases/download/v1.2.1.2/Crow-1.2.1-Linux.tar.gz>
- **Installation**: Headers installed to `/usr/include/asio/` and `/usr/include/crow/`
- **Pkg-config**: Provides `asio.pc` and `crow.pc` for easy integration

crow-test (example)

- **Recipe**: `recipes-example/crow-test/crow-test_1.0.bb`
- **Description**: Example web server application using Crow C++
- **Features**:
 - Simple HTTP endpoints
 - JSON API response
 - Health check endpoint
 - Multithreaded server

Adding to Your Build

1. Add the layer to your build:

```
bitbake-layers add-layer meta-dir/meta-crow-asio
```

2. Build the dependencies:

```
bitbake crow-asio
```

3. Build the test application (optional):

```
bitbake crow-test
```

Using in Your Applications

With pkg-config (recommended):

```
find_package(PkgConfig REQUIRED)
pkg_check_modules(ASIO REQUIRED asio)
pkg_check_modules(CROW REQUIRED crow)

target_include_directories(your_app PRIVATE
    ${ASIO_INCLUDE_DIRS}
    ${CROW_INCLUDE_DIRS}
)
```

Manual include paths:

```
target_include_directories(your_app PRIVATE
    /usr/include/asio
    /usr/include/crow
)
```

In your recipe:

```
DEPENDS = "crow-asio"
```

Testing

After building and deploying crow-test:

1. Run the test server: `crow-test`
2. Test endpoints:
 - `curl http://localhost:8080/` - Simple greeting
 - `curl http://localhost:8080/json` - JSON response
 - `curl http://localhost:8080/health` - Health check

License

- ASIO: Boost Software License
- Crow: BSD 3-Clause License
- This layer: MIT License