
GHDL Sample Project Documentation

Release 0.0.1

Pedro Cuadra

Sep 27, 2017

Contents

1	Install Dependencies	3
2	Indexing VHDL Files	5
3	Run Tests	7
4	Building System Description	9
4.1	Test Building System	10

This project provides an easy but power full building system for VHDL code to run simulations using GHDL project.

CHAPTER 1

Install Dependencies

In order to use this sample project and building system you need first to install GHDL. To install GHDL follow [this](#).

CHAPTER 2

Indexing VHDL Files

In order to GHDL to work it has to index all *entities* from all VHDL source files. For doing so simply run;

```
cd <your-project-code-path>
mkdir build
cd build
cmake ..
make index
```


CHAPTER 3

Run Tests

To build the test's binaries you first need to;

```
cd <your-project-code-path>/build  
make check
```

The summary of the tests will be displayed and it looks like the following;

For every test there's one VCD trace file automatically created. You can find the VCD files at *<your-project-code-path>/build/trace/*. The traces are placed according to the folder structure of the *<your-project-code-path>/test/* directory.

Building System Description

Our code is completely *IDE* independent. For supporting the compiling and simulation of the code we had to develop our own building system using *CMake*.

The building system is based on two simple yet powerful *CMake* macros.

```
# Add source directory
add_sources_directory(dir1 dir2 ...)

# Add source file
add_sources(file1 file2 ...)
```

The basic idea behind our building system is to specify directories having source codes and inside every directory with sources specify which files are source files. For example, `<your-project-code-path>/src/CMakeLists.txt`, specifies *common*, *decoder* and *encoder* as sources directories as follows;

```
# Add source directory
add_sources_directory(adder/
                    viterbi_encoder/)
```

Furthermore, `<your-project-code-path>/src/viterbi_encoder/CMakeLists.txt` should look like;

```
# Add sources
add_sources(viterbi_encoder.vhdl
           next_state_lkup_table.vhdl
           output_lkup_table.vhdl)
```

Note:

- You can specify source code directories and files in the same *CMakeLists.txt* file.
- You can specify as many source code directories and files using the same macro call (*add_sources_directory(...)* and *add_sources(...)* respectively).
- Every source code directory added using *add_sources_directory(...)* has to contain a *CMakeLists.txt* file.

Our main *CMakeLists.txt* at *<your-project-code-path>* does the rest.

Test Building System

For tests we have a simliar building system but the CMake macros are named differently. They are the following;

```
# Add test source directory
add_test_sources_directory(dir1 dir2 ...)

# Add test source file
add_test_sources(file1 file2 ...)
```