

# Running SoC Tests

AhaM3SoC Tests are organized using **mflowgen**. The mflowgen design name is **test-soc** and is located within the **AhaM3SoC** repo.

The **test-soc** mflowgen design has dependencies on the following repos:

- Garnet (on github)
- AhaM3SoC (on github)
- AhaM3SoC\_ARMIP (on local gitlab)

## Software Setup

The following software tools are used:

- ARM GNU GCC Compiler (arm-none-eabi-x): This is already installed on the ARM machine at `/opt/gcc-arm-none-eabi-8-2018/bin`. This should be added to the PATH environment variable
- Synopsys VCS (can be enabled with `module load vcs` on the ARM machine)

## Test Setup

The **test-soc** mflowgen design folder contains a `construct.py` file in which parameters have to be properly set.

- `ARM_IP_DIR` : refers to the location of the cloned **AhaM3SoC\_ARMIP** repo
- `AHA_IP_DIR` : refers to the location of the cloned **AhaM3SoC** repo
- `GARNET_DIR` : refers to the location of the cloned **Garnet** repo

Note:

For Garnet, make sure `pip install -r requirements.txt --upgrade` works within your virtual environment before running the tests

## Running Tests

1. clone mflowgen repo (<https://github.com/cornell-brg/mflowgen>)
2. install mflowgen : `cd mflowgen && pip install -e .`
3. create a build folder: `mkdir build && cd build`
4. Generate mflowgen build files: `mflowgen run --design <path to test-soc design folder>`
5. Find out which steps to run: `make list`

Note

- each test has a dedicated `build` and `run` steps
- there is a reduce-node called `verdict` that will collect results from all tests and report status

## Editing Tests

All tests are located in `ARM_IP_DIR/software/testcodes` . In there, there is a directory for each test.

To build a test in sandbox,

1. `cd` into the test directory
2. run `make`

`make clean` will clean up the generated artifact

## Adding a New Test

1. copy an existing test ( `hello_test`) to the new test directory
2. rename both the test folder name and the name of the test C file (both of these must shared the same name)
3. modify the test name inside the `Makefile` : (set `TESTNAME = <name of test>` . Must be same as C file name of the test)
4. Add the test name to the test list in `construct.py` (in **test-soc** mflowgen design)
5. Re-run mflowgen to re-generate the new build files