

Elementos de Sistemas - Instalando Infraestrutura

Rafael Corsi - rafael.corsi@insper.edu.br

Março - 2018

! Tempo previsto de execução **2h00**

A disciplina de elementos de sistemas faz uso de diversas ferramentas para compilar códigos em VHDL, JAVA além de automatizar o processo de testes de códigos (tanto em HDL quanto em JAVA).

Os softwares que serão utilizados são :

- Intel Quartus : Software da Intel/ Altera utilizado para programar FPGAs (compilador, IDE, ...).
- ModelSim : Software da MentorGraphics utilizado para simular HDL (VHDL/Verilog)
- Python 3 : Para automatizar os testes
- Java JDK 8 : Java Development Kit
- Editor de texto : Editor de texto para programação em VHDL/Java
- IDE de programação JAVA : Eclipse / IntelliJ IDEA

Linux (ubuntu LTS 16.04)

Para os usuários de Ubuntu lts 16.04 os seguintes passos devem ser realizados :

1 - Python 3

Execute no terminal :

```
$ sudo apt-get update  
$ sudo apt-get -y upgrade
```

Verifique se a instalação está correta executando :

```
$ python3 -V
```

Deve resultar em :

```
$ Python 3.5.2
```

2 - pip

Instale o pip (gerenciamento de pacote python) executando o comando no terminal a seguir :

```
$ sudo apt-get install -y python3-pip
```

Verifique se o pip foi instalado corretamente :

```
$ pip -V
```

Instalando as dependências

Faça a atualização do fork e note a existência do arquivo *requirements.txt*, esse arquivo possui todas as dependências python utilizadas no projeto :

Abra o bash na pasta do projeto e execute :

```
$ pip install --user -r requirements.txt
```

3 - Quartus Prime e ModelSim

Requerimentos

Execute no terminal os comandos a seguir, o Quartus necessita de dependências da arquitetura i384.

```
$ sudo dpkg --add-architecture i386
```

```
$ sudo apt-get update
```

```
$ sudo apt-get install libxft2:i386 libxext6:i386 libncurses5:i386
```

Quartus

Faça o download dos arquivos a seguir (salve na mesma todos os arquivos):

- Quartus Lite : http://download.altera.com/akdlm/software/acdsinst/17.1std/590/ib_installers/QuartusLiteSetup-17.1.0.590-linux.run
- ModelSim : http://download.altera.com/akdlm/software/acdsinst/17.1std/590/ib_installers/ModelSimSetup-17.1.0.590-linux.run
- Cyclone V (Chip usado no curso) : http://download.altera.com/akdlm/software/acdsinst/17.1std/590/ib_installers/cyclonev-17.1.0.590.qdz

Abra o terminal na pasta que os arquivos foram salvos e execute os dois comandos a seguir :

```
$ chmod +x QuartusLiteSetup-17.1.0.590-linux.run
```

```
$ ./QuartusLiteSetup-17.1.0.590-linux.run
```

! Salve o caminho na qual o **Quartus** foi instalado, ele será utilizado na próxima etapa.

3a - Configurando variável de ambiente para o VUNIT

No terminal abra execute o comando a seguir :

```
$ gedit ~/.bash_profile
```

Adicione ao final do arquivo as seguintes linhas,

```
export ALTERAPATH=~/intelFPGA_lite/17.1
export PATH=$PATH:${ALTERAPATH}/quartus/bin
export PATH=$PATH:${ALTERAPATH}/modelsim_ase/linuxaloem/
export VUNIT_MODELSIM_PATH=${ALTERAPATH}/modelsim_ase/linuxaloem/
export VUNIT_SIMULATOR=modelsim
```

Se você alterou o caminho de instalação na etapa do Quartus, deve modificar a primeira linha inserindo o caminho da instalação.

Validando

! Reinicie o computador (ou máquina virtual) para concluir a instalação

Atualize o repositório do projeto Z01 com o upstream (tutorial: B-Ferramental.pdf), deve aparecer uma nova pasta chamada de 0-TesteInfra, nesse pasta teremos scripts para validar se a instalação das ferramentas foi feita com sucesso.

Vá para a pasta a raiz do repositório Z01 e execute o seguinte comando no terminal :

```
python3 0-TesteInfra/testeVHDL.py
```

O resultado deve ser o seguinte :

```
Starting lib.tb_and16.all
pass (P=1 S=0 F=0 T=1) lib.tb_and16.all (1.0 seconds)

==== Summary =====
pass lib.tb_and16.all (1.0 seconds)
=====
pass 1 of 1
=====
Total time was 1.0 seconds
Elapsed time was 1.3 seconds
=====
All passed!
→ Z01 git:(master) x
```

Figure 1: Resultado validação infra