



Summer Fellowship Report

On

Python Based UI for OpenModelica Server

Submitted by

Sridhatta Jayaram Aithal

Under the guidance of

Prof.Kannan M. Moudgalya
Chemical Engineering Department
IIT Bombay

August 9, 2021

Acknowledgment

The fellowship opportunity we had with the FOSSEE Team, IIT BOMBAY, was a great chance for learning and professional development. Therefore, I consider ourselves as very lucky individual as I was provided with an opportunity to be a part of it. I am also grateful for having a chance to meet so many wonderful people and professionals across the country who led us through this internship period. I am using this opportunity to express our deepest gratitude and special thanks to Prof. Kannan M. Moudgalya, head of FOSSEE team, IIT Bombay, for giving us an opportunity to be a part of this project.

It is my radiant sentiment to place on record my best regards, deepest sense of gratitude to my mentors, Mr. Rupak Rokade and Mr. Sudhakar Kumar for their continuous support which were extremely valuable for my study both theoretically and practically and helping us to learn a lot many things.

I perceive this opportunity as a big milestone in my career development. I will strive to use gained skills and knowledge in the best possible way, and I will continue to work on their improvement, in order to attain desired career objectives. Hope to continue cooperation with all of you in the future.

Contents

1	Introduction	3
1.1	About Python	3
1.2	About Bash Script	3
2	Implementation of the GUI Client	4
2.1	Automated server script using BASH	4
2.2	Python GUI Client for Openmodelica server	4
2.2.1	Process Flow	4

Chapter 1

Introduction

1.1 About Python

Python is an open source language, High-level general-purpose programming language. It is a object oriented language which helps the programmers in writing the code in a very lucid way. Python has developed a platform which supports many tasks due to its high level interpretability It was built by Guido van Rossum in 1991.

For the Implementation of the UI in this implementation Python is extensively used as it supports many UI libraries modules that are already present which are required for the implementation of the use case. Python GUI library tkinter is used for creating the client for the model of the openmodellica in which the server is running at the back end.

The client consists of the GUI part where the user can view the server variables and can subscribe to the read and write part of the server variable where actually the backend controller in client can help the controlling and tapping of the values to the desired steady state using the input variable.

The User needs to input the address of the server and after observing the sequence of variable which the user wants to subscribe for input and output has to be given in the GUI client which will then try to apply the controller logic and also will record the value and plot the graph and also will be able to retrieve the values in .csv format.

1.2 About Bash Script

BASH refers to Bourne-Again Shell is a Unix shell program associated with the scripting language. It is the default part of GNU operating System Linux. It provides the varied support for command line editing, Autocompletion, Aliases, Functions, Arrays, Job Control, Timing, Command History. Here it is used for the automating the server initialisation in the command prompt as it helps in reducing the writing the commands in the terminal shell for starting the server.

Chapter 2

Implementation of the GUI Client

2.1 Automated server script using BASH

To launch the server of the openmodelica the bash script was used so that instead of writing the list of series of sequence of commands can be executed using only a single command which can be used to launch the server.

2.2 Python GUI Client for Openmodelica server

The Server runs on the dedicated IP address(Local IP) and Port address(4841). The Server 'starts' when it is pinged by the client with run bit set to '1' the server starts executing with accepting the input and giving the output with fix number of stop time specified to it. The GUI client is built in python using tkinter GUI module. The GUIClient accepts the IP address of the server and establishes the connection with the server initially to pull out the server variables containing the sequence of variables where the user has to provide them as input to the read and write variable in the GUI. When the user provides the read and write variable and setpoint which the desired output is to be as input and clicks on simulate on the GUI Client the run bit is enabled the server starts execution and starts providing the output value which is recieved as read variable and the controller is provided as the system which provides the value for the server through the write variable(keeping setpoint as a goal). The resulting output is stored as record in csv file format and the graph is also plotted at the end which is shown in the figure.

2.2.1 Process Flow

The Process flow is as follows:

1. Launching the Server with bash script to start the server.
2. Next step is to execute the code of the GUI client which launches the GUI of the simulation.
3. The User has to enter the IP address in order to get the sequence of the read and write variable.
4. After getting the sequence of the server variables that of read and write variables and the user will feed this as inpput in the GUI and also the setpoint the user desires

the output to be at click on the Simulate.

5. The run bit will be enabled the process starts for the desired time period and after the process is completed the output will be stored in a csv file and also the graph will be plotted.

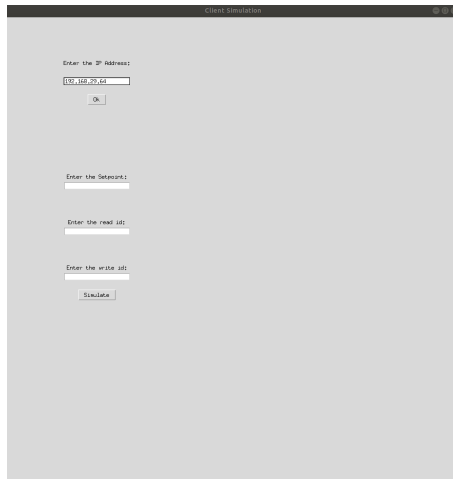


Figure 2.1: GUI Client Window

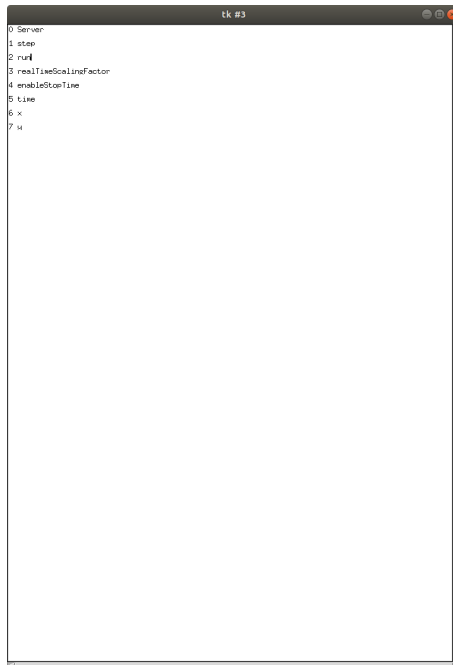


Figure 2.2: Displaying the Server variables

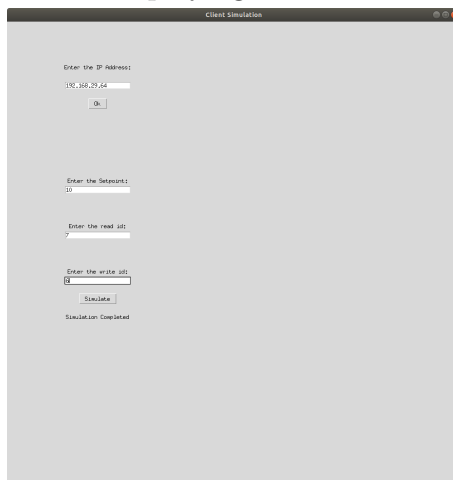


Figure 2.3: GUI Client Window after Simulation completed

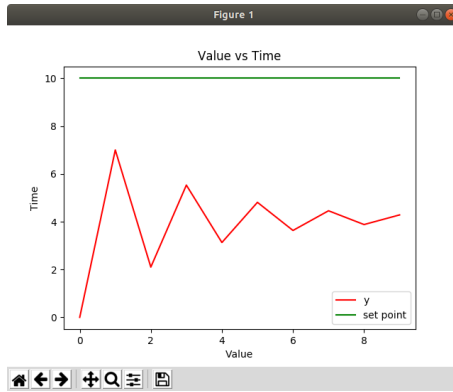


Figure 2.4: Graph of output variable and desired setpoint

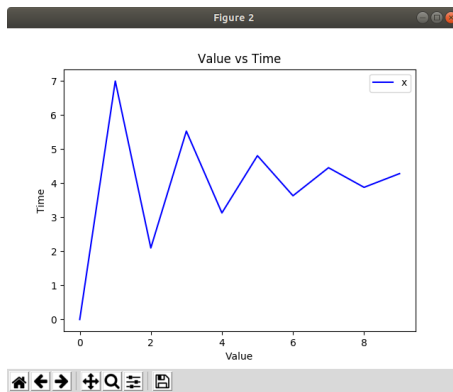


Figure 2.5: Graph of input variable

Figure 2.6 is a screenshot of a spreadsheet application showing a CSV record. The spreadsheet has columns labeled A through F and rows numbered 1 through 19. The data in column A is as follows:

Row	Column A	Column B	Column C	Column D	Column E	Column F
1	x,y,setpoint					
2	0,0,0,0,10					
3	1,7,0,7,0,10					
4	2,2,09999999999999999999,2,09999999999999999999,10					
5	3,5,53,5,53,10					
6	4,3,12899999999999999999,3,12899999999999999999,10					
7	5,4,8097,4,8097,10					
8	6,3,63320999999999999999,3,63320999999999999999,10					
9	7,4,456753,4,456753,10					
10	8,3,88027289999999999999,3,88027289999999999999,10					
11	9,4,26380897,4,26380897,10					
12						
13						
14						
15						
16						
17						
18						
19						

Figure 2.6: CSV record of the Input and output server variable

Reference

- <https://docs.python.org/3/library/tkinter.html>
- [https://en.wikipedia.org/wiki/Python\(*programming language*\)](https://en.wikipedia.org/wiki/Python_(programming_language))