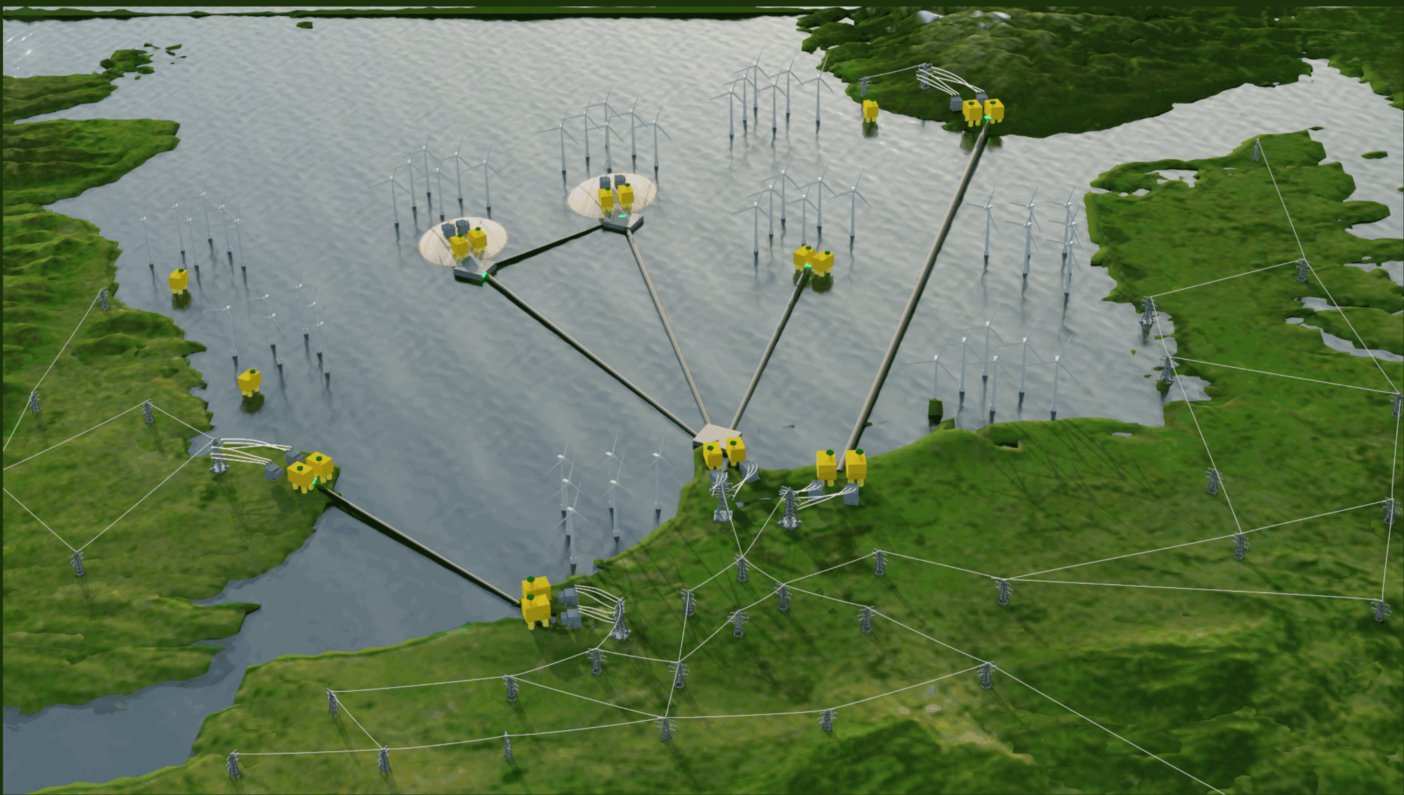
A.D Shetgaonkar

Enhanced Real-Time Multi-Terminal HVDC Power System Benchmark Models

Description and instruction guide.



TUDelft

Contents

1 Introduction

This document will provide step-by-step instructions on using the models provided in the git directory.

1.1 RTDS Hardware requirements

In order to run this model, the user needs to have the following hardware requirements in Table 1. Furthermore, this model can also be run on the old generation core (i.e., Pb5)

Table 1 Core requirements for different HVDC models

Networks	Cores
“CIGRE_2TERMINAL_525KV_RTS_MODEL_3D_check.rtfx”	3
“CIGRE_3TERMINAL_525KV_RTS_MODEL_3D_check.rtfx”	4
“CIGRE_4TERMINAL_525KV_RTS_MODEL_3D_check.rtfx”	4
“CIGRE_5TERMINAL_525KV_RTS_MODEL_3D_check.rtfx”	7

1.2 RSCAD version

In order to run the models, the user needs to upgrade the RSCAD version to the latest one available today (July 2023), i.e., Fx 2.0.

1.3 Download and get ready.

The following steps need to be followed in order to run the files:

Step 1: Download the files from the git directory.

Step 2: Extract the zip “HVDC-RTDS-Models.zip” in a temporary folder.

Step 3: From the extracted folder, copy the “CSE_MODEL_FX2.0” folder, and paste it to your RSCAD user directory (example: “C:\RSCAD\RTDS_USER_FX\fileman\”).

Step 4: Now open the RSCAD FX2.0, and load the desired models from the CSE_MODEL_FX2.0 folder from your directory. For this explanation, I am considering the “CIGRE_5TERMINAL_525KV_RTS_MODEL_3D_check.rtfx” test case.

Step 5: Upon loading the above file, you will see the following *Draft* tab as shown in Figure 1:

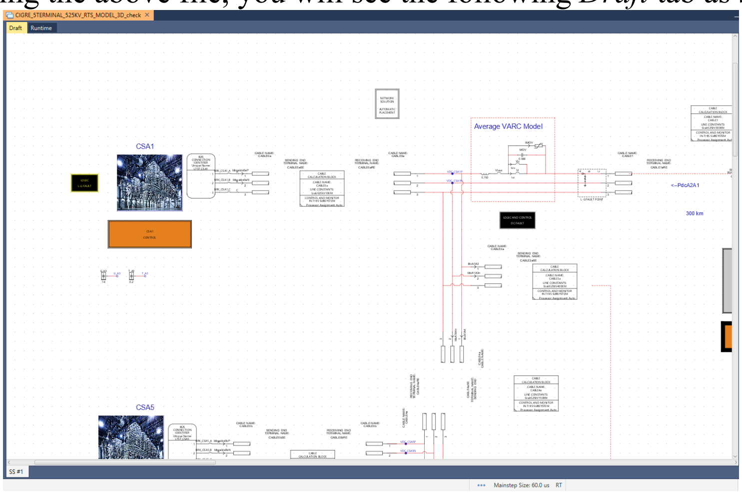
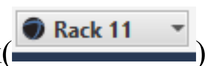


Figure 1 Screenshot of opened “CIGRE_5TERMINAL_525KV_RTS_MODEL_3D_check” file

Step 6: Compile the file () on a rack(

) with suitable cores, in this case, a minimum of seven cores.

Step 7: Now go to the *Runtime* tab, and you will see the following image, as shown in Figure 2.

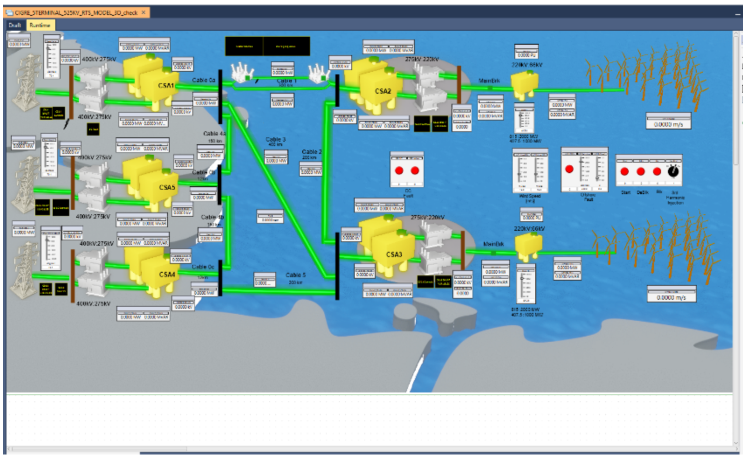


Figure 2   Screenshot of Runtime for" CIGRE_5TERMINAL_525KV_RTS_MODEL_3D_check "file.

Step 8: Now open and run the script "*Parameter_settting*". This will configure the draft variables in the *runtime.*

Step 9: Now open and run the script "starting_squence_5T_file". This will start the simulation with a starting sequence mentioned in the paper. Once the script is completed now, you can begin your study.

2 Draft description.

In this section, we will describe the draft file of an example file. The highlight of this section will be an overview of the different elements of the draft file. The description of the following remains the same for the rest network.

2.1 CIGRE_5TERMINAL_525KV_RTS_MODEL_3D_check draft file.

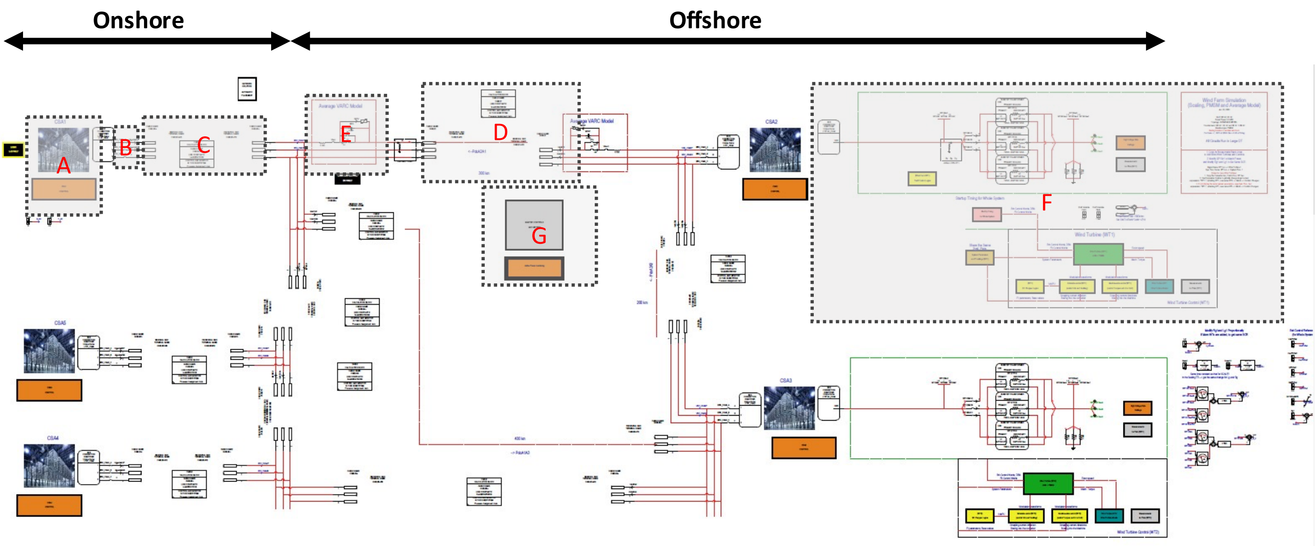


Figure 3 draft Screen-short of "CIGRE_5TERMINAL_525KV_RTS_MODEL_3D_check draft file" draft file.

In Figure 3, the **A** region indicates the converter station and connection between *main* and *small timestep*. It also consists of control & protection., meter, and draft variables. In the mentioned network, we have five such stations.

The **B** region indicates the disconnector, which dis/connects the converter station from the land and sea land cables. In the above network, we have five such disconnectors. The **C** region is a DMR (Dedicated Metallica Return) land cable with a length of 20 km. This cable connected the onshore converter to the sea cable. In the above network, we have three such land cables. The **D** region is submarine cable. Considering the mentioned network, we have six cables. The length of these cables can be found in the paper.

E region indicates the VARC DC CB. This region also consists of fault logic and the operating time of VARC. In all the models, the VARC is connected to a positive pole and placed at either end of the cable.

Region **F** consists of Wind turbines and scaling transformers to mimic the wind power pack/wind farm. We have two such wind power packs. Region **G** has general meters and master controls.

2.1.1 Converter station (Region A)

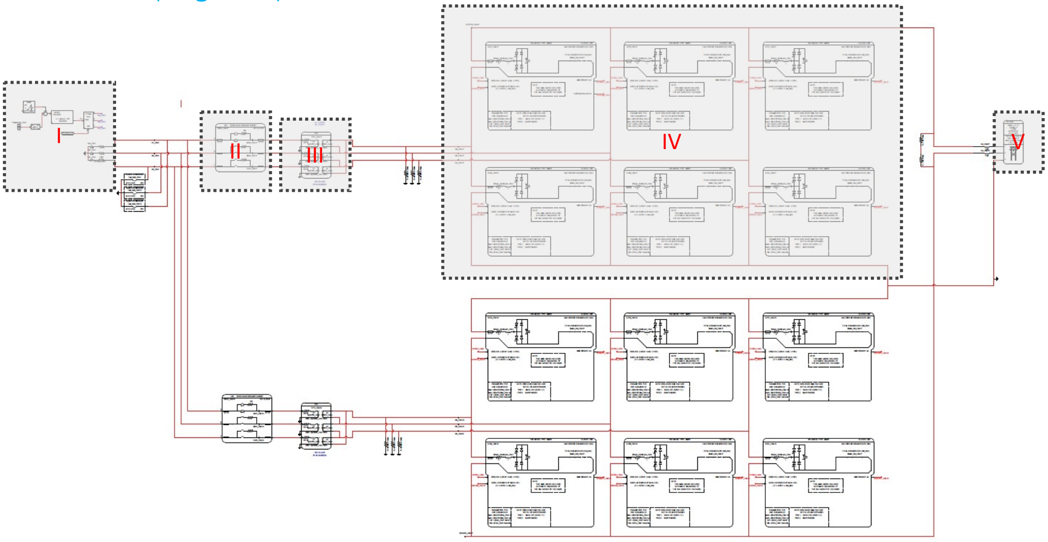
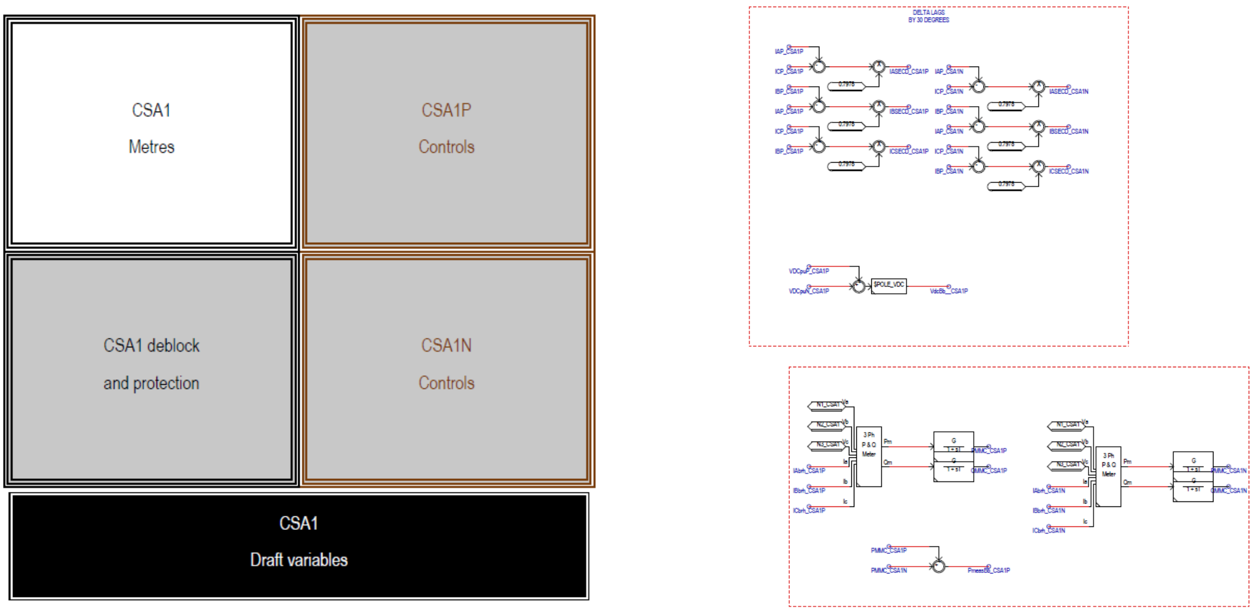


Figure 4 draft Screenshot .of a small timestep environment of an onshore converter station

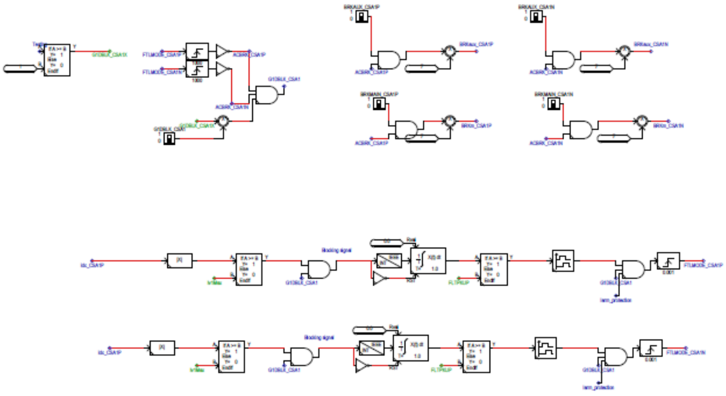
The converter station is built in a *small timestep* environment. The converter station has five regions for positive poles.
Region **I** is an AC system with grid impedance for SCR impact. The II region consists of the pre-insertion resistor. Region **III** is a star-delta transformer. Region **V** is the connection between a *small timestep* to *main step.*
Due to the bipolar topology, we have an identical negative pole setup except for regions **I** and **V**. We do not have region **I** for the offshore network. Furthermore, we have two **V** regions.

In the control section of the converter station (Figure 5 (a)), we have five blocks, indicating two controls (positive and negative pole), meters, protection, and Draft variables.

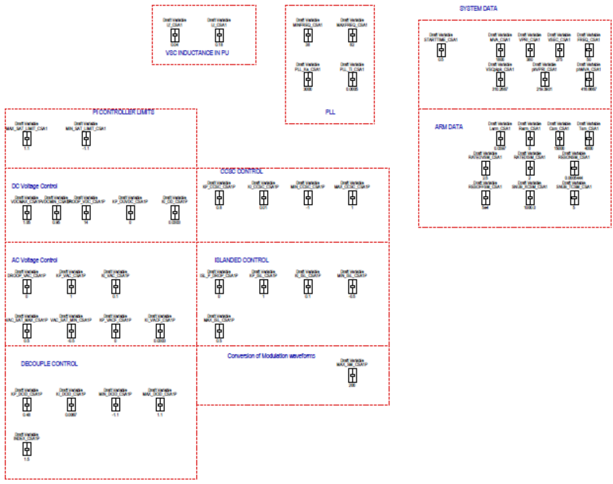


(a) Control, protection, meter, and draft variables block of the converter station

(b) Meter section of the converter station



(c) Protection section of the converter station



(d) draft section of the converter station

Figure 5 Control, protection, meter, and draft variable section of the converter station

Furthermore, Figure 6 indicates the screenshot of control schematics of the controls for a pole of the converter station. This section comprises Outer voltage control, inner current control, and lower-level control. Also, this control scheme has sequence control and islanded mode control. In addition to this, it also has circulating current suppression control.

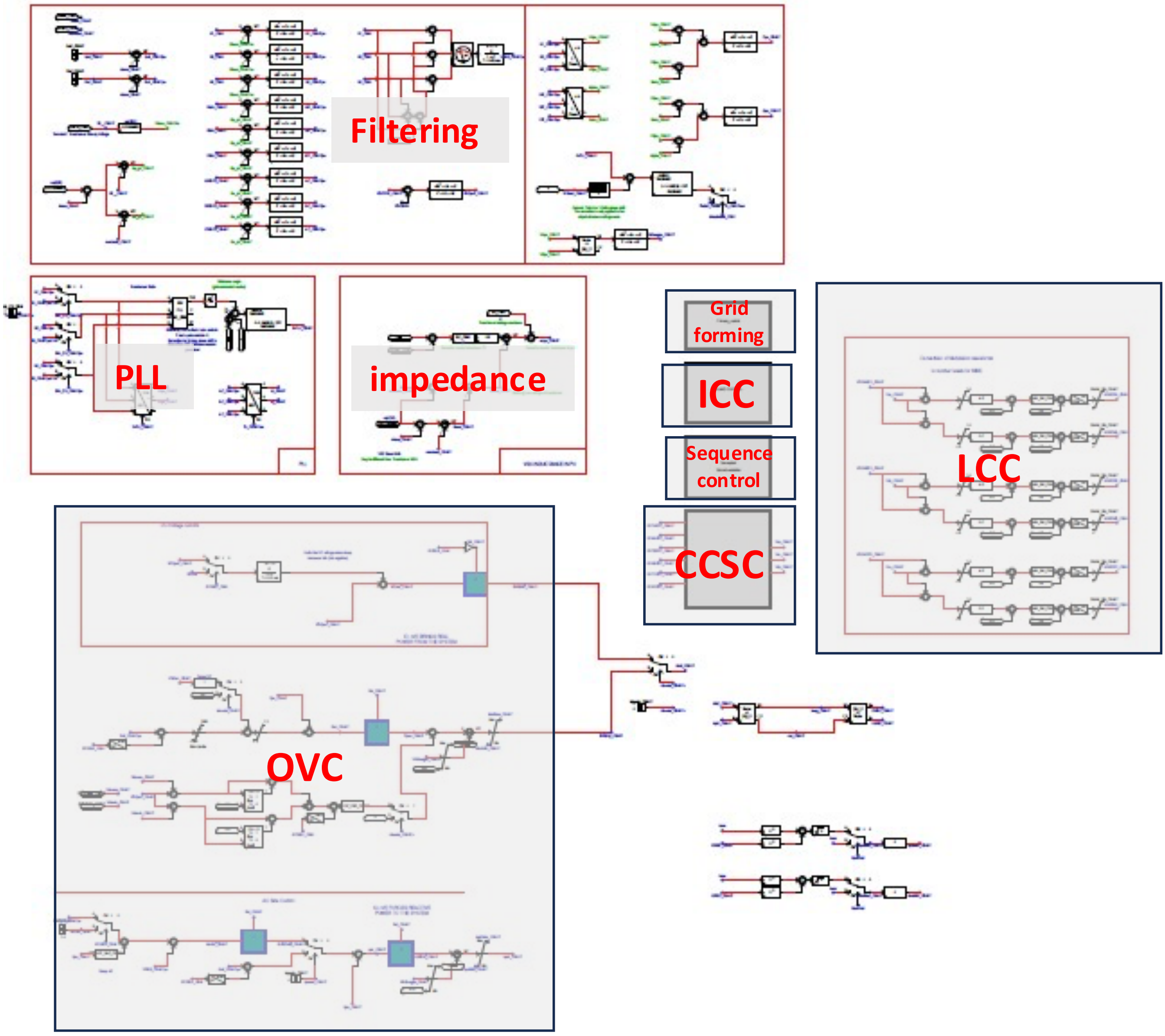


Figure 6 Screenshot of the control schematic of the converter station.

2.1.2 Wind Power Pack/Wind Farm (Region F)

For detailed information about wind turbine modeling and control, readers are requested to read the example section of RSCAD [1] and the paper. Since the material for the wind turbine is well-written, we are skipping this readme.

3 Runtime description.

3.1 CIGRE_5TERMINAL_525KV_RTS_MODEL_3D_check model

The Figure 7 shows the screenshot of the *runtime*. In this *runtime,* we have six regions, Region **A** indicates the control functions & setpoints, and Draft variables. These two blocks in this region remain the same for all the converters. Furthermore, region **B** indicates AC fault duration, fault button, and re-set signal near the *CSA1* converter. Region **C** has two hierarchy boxes, namely *Master control* and *Starting sequence.* As the name indicates, the *Master control* has all necessary global setpoints and control functions. Similarly, *Starting sequence* has control switches that give the breaker close command and de-block signal to the converter station.

Region **D** consists of wind turbine controls, draft variables, and setpoints. Furthermore, region **F** consists of a wind speed control slider, offshore faut & fault duration, and Energisation buttons of the wind turbine. Region **G** consists of DC fault and reclose command for DC breakers.

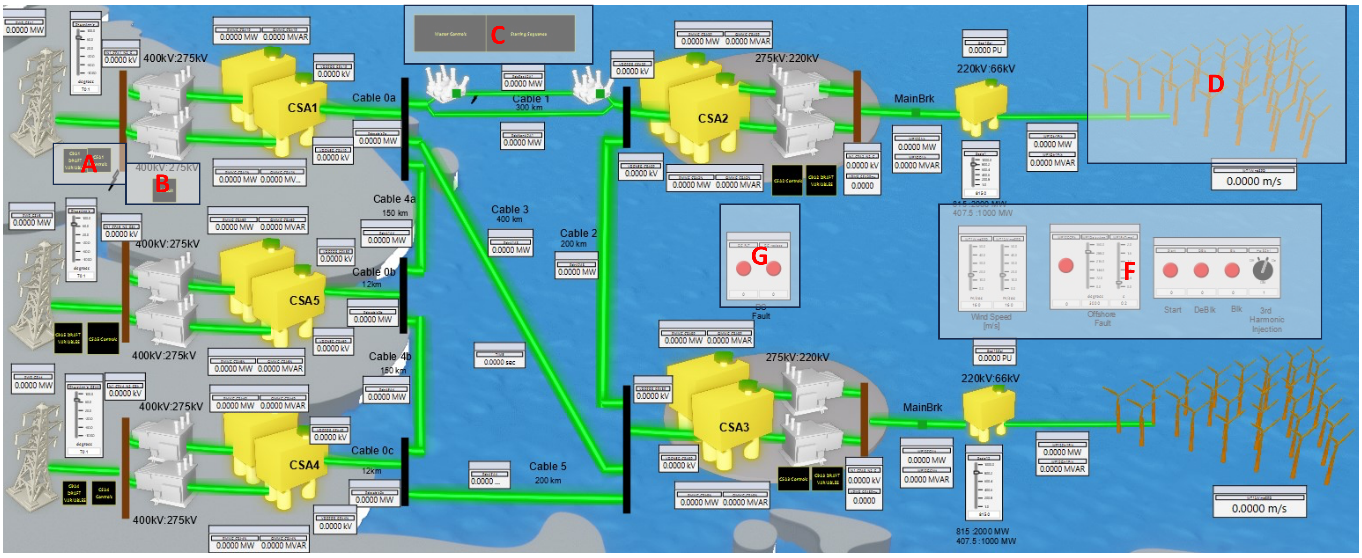


Figure 7 Screenshot of the runtime

4 SiL setup for wind data collection.

In this section, we will explain the SiL setup between wind data collection from the wind probe in the North Sea via the world wide web.

4.1 Python code short description.

The Python script called "Real_time_wind_co_simulation.py" connects RTDS to python and collects the data using TCP/IP connection.

The provided code is a Python script that uses the Selenium library to automate web browser interactions, collects data from web pages, and visualizes the data using Matplotlib.

Here's a breakdown of the code:

1. The script imports the necessary libraries: `*selenium*`, `*winsound*`, `*socket*`, `*time*`, `*matplotlib*`, and `*numpy*`.
2. There is a boolean variable `*plotting*` set to `*True*`, indicating that data will be plotted.
3. The script defines a function `*move_figure*` that moves the position of a matplotlib figure on the screen.
4. The `*on_launch*` function sets up the initial plot with four subplots (`*ax*`, `*ax1*`, `*ax2*`, `*ax3*`) and initializes the lines to be updated later.
5. The `*on_running*` function updates the data in the plot by modifying the lines and axes.
6. The script creates two instances of the Chrome WebDriver using Selenium, sets their window size and position, and navigates to specific web pages.
7. The script establishes a TCP socket connection to a specific IP address and port.
8. If `*plotting*` is `*True*`, the script initializes variables and creates a matplotlib figure and subplots using the `*on_launch*` function.
9. A while loop starts, and within it, data is extracted from web pages using Selenium, processed, and sent via the socket connection.
10. The extracted data is printed to the console.
11. The extracted data is manipulated and injected into the socket connection for further processing.
12. The web pages are refreshed, and there is a short delay before the next iteration of the loop.
13. If `*plotting*` is `*True*`, the extracted data is appended to the respective data arrays, and the `*on_running*` function is called to update the plot.

The code essentially automates the collection of wind speed and active power data from two web pages, communicates the data via a socket connection, and visualizes the data in real-time using Matplotlib if `*plotting*` is set to `*True*`.

4.2 Step to follow for SiL

Follow the nine steps explained in section 1.3. After that, follow these steps:

Step 10: Open and run the "*ListenOnPort*" script.
Step 11: Now open and run the Real_time_wind_co_simulation.py. Upon running, you should get the following results as shown in Figure 8

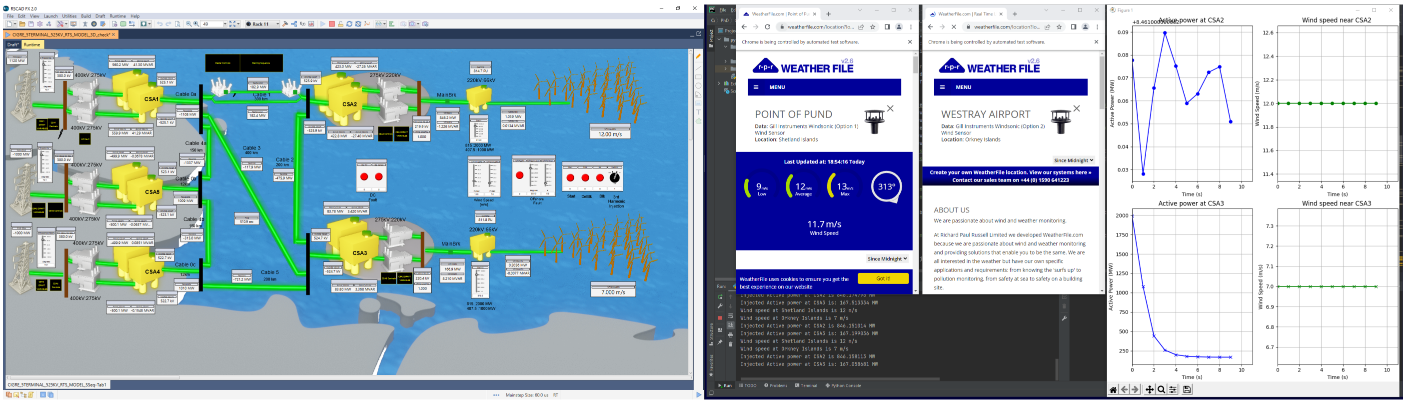


Figure 8 Screen short of SiL setup

5 Duplicate the converter station.

In order to duplicate the converter station. You need to follow the following steps.

Step 1: Group the region **A** in Figure 1 by pressing "Ctrl+g" on your keyboard.
Step 2: Copy and paste the region **A** in Figure 1.
Step 3: Right-click on the pasted new region, go to *edit* > *expression*
Step 4: In the Expression tab, under *Expression* input, write the existing Converter station name example "*.*CSA4.**". Now in *To Replace (substring)* input, give the existing Converter station name example "*CSA4*".
And in the *Replace With* input give the new Converter station example "*CSA5*". Then Click on *Find Matches* button. It will show all the matched cases.
Step 5: Click on *Apply* button. Now you have successfully duplicated the converter station.

Note: to duplicate the wind farm, follow the instruction provided by RSCAD example case of Renewables[1]**.**

6 Bibliography

[1] RTDS Technologies, “Standardization of Renewable Energy System Modelling,” 2022.