VSC – HVDC Modelica Models for Power System Time Domain Simulation

Md Jahidul Islam Razan

KTH Technology and Health

Master of Science Thesis in Electrical Engineering
Stockholm 2014

VSC – HVDC Modelica Models for Power System Time Domain Simulation

VSC – HVDC Modelica Models för Tidssimulering av Elkraftsystem

Md Jahidul Islam Razan

Master of Science Thesis in Electrical Engineering
Advanced level (second cycle), 30 credits
Supervisor at KTH: Francisco José Gómez
Examiner: Luigi Vanfretti
School of Electrical Engineering
XR-EE-EPS 2014:008

Royal Institute of Technology
Tecknikringen 33
SE-100 44 Stockholm, Sweden
http://www.kth.se/sth

Acknowledgement

First of all I would like to express my gratitude to EIT KIC-InnoEnergy for giving the opportunity and all sorts of support to pursue this prestigious program of Smart Electrical Networks and Systems (SENSE). This research project would not have been possible without the support and encouragement of many people. I take this opportunity to express gratitude to the people who have been instrumental in the successful completion of this project. First, I would thank Dr. Luigi Vanfretti, head of the "Smart Transmission Systems Lab" and examiner of the thesis, for allowing me to work with this research project. I would like to express my deepest gratitude to my supervisor Francisco Jose Gomez, who abundantly helped me and offered me his invaluable assistance, support and guidance in various parts of the project. I wish to express gratitude to my fellow students Mohammed Ahsan Adib Murad, Omar Kotb and Luis Diez for sharing their valuable thoughts with me in solving various problems. I would take this opportunity to express my love and gratitude to my beloved family members for their understanding & endless love, throughout the duration of my studies. Also I would like to thank the program coordinator of SENSE program Dr. Hans Edin for his valuable support during the whole study period. Finally, all praise to the almighty Allah (SWT), without His mercy nothing would have been possible.

Abstract

Voltage Source Control (VSC) technology in High-Voltage Direct Current (HVDC) transmission can be used for bulk power transmission, asynchronous network interconnections, back-to-back AC system linking, and voltage/stability support. Besides, it can be used to control the active and reactive power independently without the addition of extra compensating equipment. The implementation of these kinds of models is now specific for Electro Magnetic Transient simulation software like EMTP-RV. However, these models can therefore be defined in a standardized language that allows exchange of models between various modeling and simulation tools (supporting the standard) and therefore providing a clear separation between the model and the solver. This allows uniformity in modeling tools not only in terms of parameters, but also in case of explicit equations.

Equation based modeling languages like Modelica can offer aid to resolve to realize an open implementation of these models, to provide a standard implementation and make them more easily to share across other different simulation tools. Modelica is an object oriented language which offers the feature of equation based modeling of physical systems as well as its components. Also in Modelica it is possible to build new components and reusable libraries as it uses the fundamental concepts of object oriented programming. Electromagnetic Transient Program (EMTP) is largely used for simulation of the transient of power system.

Due to the large number of submodules (SMs) to represent full detailed models of the MMC topology, the work has been focused on the implementation of the simplest model in Modelica, from an existing EMTP implementation.

Abstrakt

Voltage Source Control (VSC) teknik i hög Voltage Direct Current (HVDC) överföring kan användas för bulk kraftöverföring, asynkrona sammankopplingen, back-to-back AC-system länkning och spänning/stabilitetsstöd. Dessutom kan den användas för att styra den aktiva och reaktiva effekten självständigt utan tillsats av extra kompenserande utrustning. Genomförandet av dessa typer av modeller är nu specifikt för elektromagnetisk Transient simulering program som EMTP-RV. Dock kan dessa modeller därför definieras på ett standardiserat språk som tillåter utbyte av modeller mellan olika modeller och simuleringsverktyg (som stöder standarden) och därmed ger en tydlig åtskillnad mellan modellen och lösare. Detta gör att enhetlighet i modelleringsverktyg, inte bara i fråga om parametrar, utan även i fall av explicita ekvationer.

Ekvation baserade modelleringsspråk som Modelica kan erbjuda stöd för att besluta om att realisera en öppen tillämpning av dessa modeller, för att ge en standard genomförande och göra dem lättare att dela mellan andra olika simuleringsverktyg. Modelica är ett objektorienterat språk, som ger funktionen av ekvationen baserad modellering av fysiska system såväl som dess komponenter. Även i Modelica är det möjligt att bygga nya komponenter och återanvändbara bibliotek som den använder de grundläggande begreppen objektorienterad programmering. Elektro Transient Program (EMTP) är till stor del används för simulering av övergående i kraftsystemet.

På grund av det stora antalet submoduler (SM) för att representera hela detaljerade modeller av MMC topologi har arbetet varit inriktat på att genomföra den enklaste modellen i Modelica, från en befintlig genomförande EMTP.

Contents

List of Figures

List of tables:

1. Introduction

The electric power system is one of the fundamental components of modern society. It has been Alternating current (AC) that is dominating the power industry for large scale industrial and domestic uses for quite a long time. But for long distance transmission, Direct current (DC) offers multiple advantages over AC which have caused the momentum of High Voltage Direct Current (HVDC) transmission in recent years, especially in the interconnected grids [1]. Voltage Source Converter (VSC) technology has become very popular in these HVDC transmission systems for its controllability, compact and modular design. With VSC technology, it is possible to transmit large amount of power for a long distance, even through weaker networks with a significantly low level of short circuit power. Modular Multilevel Converter (MMC) topology of VSC technology has reduced the necessity of multilevel converter topologies and has improved the efficiency and resilience of the transmission network [1].

Today's transmission grid is also evolving at a faster rate due to the continuous integration of renewable energy resources and growing interconnections. It has made the system analyses significantly complex. In order to ensure the secured operation of power system, efficient tools, methods and software need to be developed. At the same time it is also important to develop sustainable methods and analysis tools in a way so that these methods and software do not get obsolete very often which will impose the task of reinventing the wheel. Existing software that performs dynamic analysis have been found to be inconsistent across various platforms. Hence, it is important to develop tools that show consistency in different simulation platforms in order to facilitate coherent work flow. A bounded solver with mathematical model is another issue that needs to be addressed to make the task of model validation easier [2]. Modelica is an object oriented language that allows equation based modeling of physical systems and various components. Modelica models can be defined in a standardized language which will facilitate the exchange of models among various modeling and simulation tools (supporting the standard) and therefore providing a clear separation between the model and the solver [3], [4].

The aim of this project is to develop an Average value model and its control system of the MMC-VSC-HVDC model in Modelica; and to validate the developed model by comparing it to an EMTP model. In particular, the most important aspect is to validate each component of the DC link (including the control system), so that they can be re-utilized in other simulation environments using the FMI standard.

2. Literature Review

2.1 Power system simulation (conventional vs Modelica)

Modern power system has become much more complex, with the growing number of interconnections between power system networks. To ensure stability in energy supply and security of a power system, complex dynamic analyses are required. There exist different numerical and modeling approaches in order to accomplish various analyses, but all these are not without limitations. One of the challenging problems that often arise is a lot of these dynamic models and simulations of the power system are inconsistent across various platforms. Hence, it is important to obtain models of power system that shows consistency in different simulation platforms in order to facilitate coherent work flow. Several factors contribute to this platform dependent behavior of models. For instance, it may be due to data formats since data format can be platform dependent. Also the components of every dynamic model are based on simplifications and assumptions which significantly differ from platform to platform. This can lead to ambiguous interpretation of how the design of the component was actually accomplished [5].

In equation based modeling, simulation of each component is open for necessary modifications. This fact leads to the opportunity of implementing new components and libraries in order to simulate the behavior of individual component as well as a full model of a system. Thus, user can build his own customer defined components and models. For unambiguous exchange of models among various platforms Modelica based tools can be used since the tools developed in Modelica are developed by using common standard language. This prevents the loss of any significant information about the exchanged models. Thus, now it has become important to focus on quality of solvers instead of the quality of the models if well-defined models are exchanged and they demonstrate inconsistent behaviors across different platforms. This inconsistency may arise due to the fact that while the model is correct the solver is not capable of simulating the model. Or the model parameters that have been used to validate the models might be incorrect and the problem lies with the model validation. This kind of problem can be resolved by getting access to the explicit model equations [6].

Modelica is an object oriented language which offers the feature of equation based modeling of physical systems as well as its components [3], [4]. Modelica effort started in mid-nineties by a group of researchers specialized in computer science. It was initiated in order to promote unambiguous and consistent modeling language for the evaluation of complex, heterogeneous physical systems in various modeling platforms. The first version of Modelica language was specified in the year 1997 and since then there have been 60 design meetings to find the possibilities of further improvement. Modelica language is a suitable simulation language that allows mathematical simulation. Modelica implements fundamental concepts of object oriented programming like packages, classes, inheritance, and components. This provides the opportunity of structuring and reusing of models. Modelica provides acausal format implementation of differential algebraic model equations, and it does not require the user to specify the direction of flow, i.e. The Modelica Standard Library implements different types of physical connectors to allow the connection between components from different domains. It is automatically determined by the component topology. This is a different approach from the block-based modeling where the input and outputs have to be specified by the user [7]. In Modelica, it is possible to use both textual and graphical modeling to implement physical systems. Also mixed continuous, discrete (hybrid behavior), user defined functions, and interfacing with external C or Fortran code is possible [8]. The reusable Modelica library components developed by a user can later be used by other people for analyzing systems having

similar components. Generally, the library developers mostly work with coding, using the laws of nature in the form of differential algebraic equations. A lot of Modelica libraries have been developed already which are available for free while some others are commercial libraries covering different areas of thermodynamics, power system, automotive to space applications. Also, there exists a free library developed by The Modelica Association [9].

2.2 High Voltage Direct Current

Historically, at the beginning commercial electricity generation and transmission was initiated in the form of DC. But DC was not very convenient to transmit over a long distance and hence alternating current took over. However, with the development of high voltage valves it is possible to transmit DC power over a long distance and that has given the rise of HVDC transmission systems. Typically HVDC is preferred as an option when large amounts of power (>500 MW) are transmitted over long distances (>500 km). Also in the case of transmitting power under water and interconnection of two AC networks in an asynchronous manner, HVDC is considered as a more viable option. HVDC combines the best features of old installations along with the recently developed technologies and materials. It has also proven to be competitive for its flexibility, efficiency and lower impact on the environment [10].

Due to skin effect AC resistance of transmission cable is higher than DC resistance and that causes higher transmission loss in AC transmission system. AC transmission line suffers more from switching surges and transient over-voltages than its DC counterpart. For stable operation under normal condition AC system is operated under a low load angle since the load angle is instantaneously affected by the disturbances. If there is no compensation load angle is also dependent on the distance which can be increased by using series capacitor that counteracts the inductive reactance of the line. In case of DC transmission system these considerations of reactance and thereby reactive power compensation, stability and distance limitations are not crucial to consider. This is why for very long distance transmission HVDC is the only suitable technical alternative. AC transmission line requires higher interspacing spacing between the lines. It takes three conductors for AC transmission system while DC needs two conductors for same power transmission capability which causes a higher cost of transmission lines for AC for the same transmission capacity.

The DC transmission line has a lower electric field problem because of lower steady state displacement current and that's why DC system requires lower right of way (ROW) and height than AC system. HVDC overhead lines have a lower radio interference level than HVAC overhead lines [11]. In case of connecting two asynchronous networks HVDC transmission provides the only option which cannot be done with two AC networks. For connecting remote offshore wind farms with the mainland grid HVDC system appears to be a better choice [12]. An optimized HVDC transmission system has lower transmission losses than AC for an equal amount of power transfer capacity. For HVDC, the converter stations contribute to the losses, but this loss only comprised of 0.6% of total loss and the total transmission loss is lower than AC transmission. Active power link of HVDC is very easy to control. There is no contribution of short circuit current from HVDC transmission system towards the interconnected AC system [13].

Again, HVDC transmission system is not without limitations. The HVDC converter substations are more complex than HVAC converter substations. Alongside the converting equipment HVDC converter substations require more complicated control and regulation system. During a short circuit fault in an AC power system close to the HVDC, substation fault also transmits in HVDC system. Grounding of HVDC transmission is more difficult as permanent contact with the earth is required for proper operation, to ensure that dangerous step voltage is not a safety threat anymore. Electro-corrosion of underground metal installations like pipelines can take place due to the flow of current through the earth in monopole systems [11]. HVDC system

requires AC to DC conversion and these converters inject harmonics and affect power quality, hamper electronic devices and can cause system oscillation [14].

2.3. Voltage Source Converter

VSC technology in HVDC transmission can be used for bulk power transmission, back-to-back AC system linking, and voltage/stability support. Hence, this technology has become an alternative energy source that is technically and economically efficient. The application of VSC topologies shows several advantages over traditional Line-commutated Current Source Converters (CSCs). Using the VSC both active and reactive power can be controlled independently without the addition of extra compensating equipment while on CSC topology the active power is dependent of reactive power supply. If required, VSC can supply or absorb reactive power from the system which is crucial to regulate the bus voltage. For VSC commutation failure is not an issue and fast commutation between the terminals is not required for control purposes. VSC converters use PWM (Pulse-width modulation) instead of fundamental switching frequency and that makes the required filter size smaller and simpler. Power reversal can be done without the voltage reversal and transformer is not needed to assist the commutation process as the converters are controlled semiconductors. By using VSC it is possible to connect weak AC networks or network without having a generation source and hence network with a very low short circuit level [1], [15].

Various conventional topologies like two-level, multi-level diode-clamped and floating capacitor multi-level converters are in use with some other new ones. Yet to keep things simple, two and three level diode clamped converters are mostly used. However, the introduction of the Modular Multilevel Converter (MMC) technology with series-connected half-bridge modules has greatly reduced the limitations of the multilevel converter topologies of HVDC applications. In MMC technology lower switching frequency reduces the converter losses and it also eliminates filter requirements. MMC technology also has a large number of sub-modules (SMs) which has made it highly scalable [16].

3. VSC-HVDC Models

MMC model is comprised of a large number of insulated-gate bipolar transistors (IGBTs) which makes the simulation difficult and time consuming. Any detail modeling of MMC modules must consider this enormous number of IGBTs and small numerical integration time steps in order to accurately depict the MMC behavior in case of switching events. This large computational time creates the necessity to develop models having smaller computational time.

MMC models have been implemented by various schemes like: (1) full detailed model, (2) detailed equivalent model, (3) switching function arm model and (4) average value model [16]. Due to the difficulties explained before, the implementation of an Average Value Model (AVM) is a suitable solution that is capable of simulating a proper response with necessary accuracies. In AVM, system dynamics are approximated by neglecting switching details which in turn enable the user to perform simulations with less computational resources in a shorter period using larger integration time steps.

In this work the full AVM model and specific controls have been developed in Modelica. The reference model for this AVM has been taken from the EMTP implementation of the model [16] that follows the CIGRE specification [17]. This model uses an Upper level control type which has also been implemented in this project. Both AVM model and Upper Level Control model have various specific components inside that perform algebraic operations. Since Modelica Standard Library (MSL) does not contain a lot of those components that perform these algebraic operations, it has been a critical task to identify their behavior and build the components to achieve the proper implementation of these models in Modelica.

3.1. AVM Model

The Average Value Model (AVM) has been designed in a way where the IGBTs are not explicitly modeled. The MMC behavior has been obtained by replacing IGBTs with controlled voltage source and controlled current source. In these controlled sources the harmonic content of the modulation control is also included in the AC waveforms [18]. The AVM model needs an Upper level control for generating three phase reference voltage quantities. Implementation of the different blocks of Upper level control is discussed in later sections.

Figure 1 shows the top level block of AVM model in EMTP which contains both the AC and the DC-side circuits.

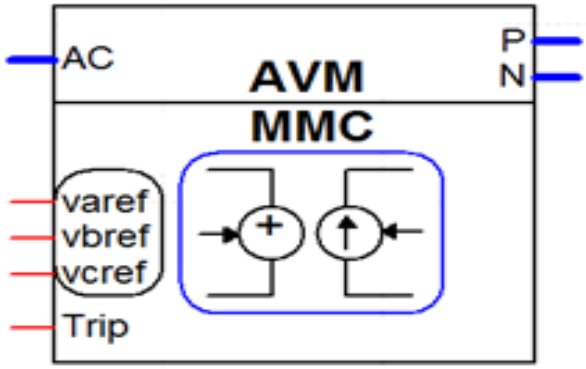

Figure 1 EMTP implementation of the Average Value Model [9]

3.1.1. AC side representation of the AVM

Figure 2 shows the circuit that represents the AC side of the proposed AVM model and the following equations can be derived from it. Equations for each phase j can be expressed as follows, where $j = a, b, c$:

$$v_{uj} = v_{uj}^{SM} - L_s \frac{di_{uj}}{dt} \tag{1}$$

$$v_{lj} = v_{lj}^{SM} - L_s \frac{di_{lj}}{dt} \tag{2}$$

$$v_{uj}^{SM} = \sum_{k=1}^{N_{arm}} \left(S_{u_{jk}} vc_{u_{jk}} \right) \tag{3}$$

$$v_{lj}^{SM} = \sum_{k=1}^{N_{arm}} \left(S_{l_{jk}} vc_{l_{jk}} \right) \tag{4}$$

$$v_j = -v_{uj} + \frac{v_{dc}}{2} = v_{lj} - \frac{v_{dc}}{2} \tag{5}$$

$$v_{uj}^{SM} = -v_j + L_s \frac{di_{u_j}}{dt} + \frac{v_{dc}}{2} \tag{6}$$

$$v_{lj}^{SM} = v_j + L_s \frac{di_{l_j}}{dt} + \frac{v_{dc}}{2} \tag{7}$$

where v_{uj} refers to the voltage of the upper arm on each phase j and v_{lj} refers to the voltage at the lower arm on each phase j. Equations v_{uj} and v_{lj} also include the voltage of the reactor, identified by L_s. The voltages v_{uj}^{SM} and v_{lj}^{SM} in Figure 1 correspond to the total voltage of all upper and lower submodules (SMs). v_{uj}^{SM} And v_{lj}^{SM} voltages are functions of the number of all capacitors represented by the equation (3) and equation (4). Binary functions $S_{u_{jk}}$ and $S_{l_{jk}}$, in equations (3) and (4), define the state of each capacitor [18].

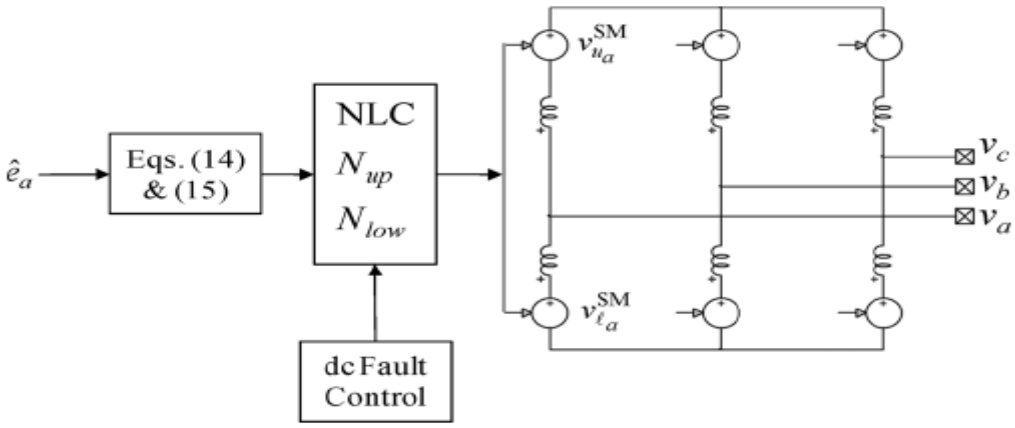

Figure 2 MMC AVM AC-side representation. Only phase -a control blocks have been shown [19]

Arm current in each phase can be described by the equation (8) and (9)

$$i_{uj} = \frac{i_j}{2} + \frac{I_{dc}}{3} + i_{zj} \tag{8}$$

$$i_{uj} = -\frac{i_j}{2} + \frac{I_{dc}}{3} + i_{zj} \tag{9}$$

where the circulating current can be expressed as:

$$i_{zj} = \frac{i_{uj+}i_{ij}}{2} - \frac{I_{dc}}{3} \tag{10}$$

$$i_{za} + i_{zb} + i_{zc} = 0 \tag{11}$$

In AVM model it is assumed that all the capacitor voltages are perfectly balanced at any time and the second harmonic circulating currents $\left(i_{zj}\right)$ are zero. By subtracting (6) from (7),

$$v_j = \frac{L_s}{2}\frac{di_{l_j}}{dt} + e_j \tag{12}$$

where,

$$e_j = \frac{v_{lj}^{SM} - v_{uj}^{SM}}{2} \tag{13}$$

Replacing the values of equation (8) and (13) into (6) give

$$v_{uj}^{SM} = -(v_j - \frac{L_s}{2}\frac{di_{u_j}}{dt}) + \frac{v_{dc}}{2} = -e_j + \frac{v_{dc}}{2} \tag{14}$$

By using the same method for the lower arm equation following can be obtained :

$$v_{lj}^{SM} = v_j - \frac{L_s}{2}\frac{di_{u_j}}{dt} + \frac{v_{dc}}{2} = e_j + \frac{v_{dc}}{2} \tag{15}$$

The AC side representation of AVM for the EMTP model is shown in Figure 3.

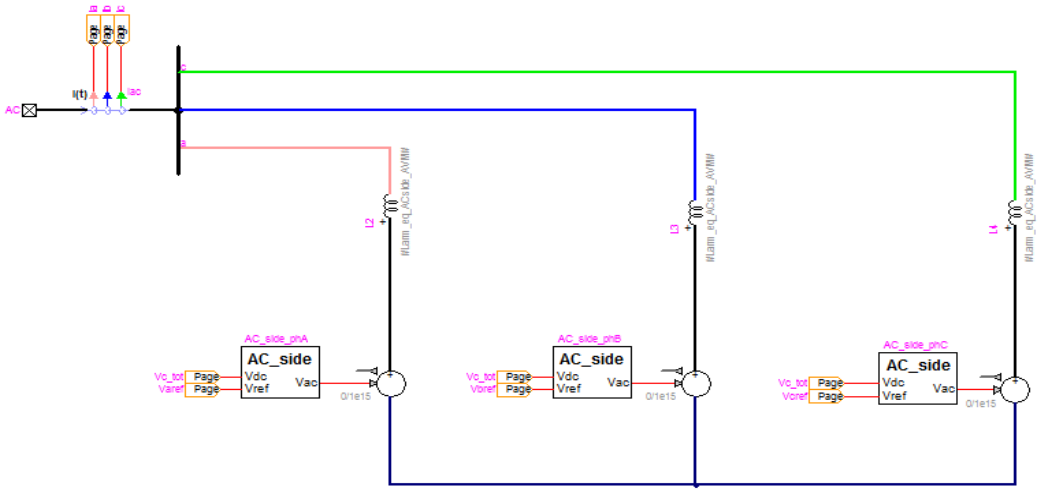

Figure 3 EMTP implementation of AC side of AVM

The Modelica implementation of the AC-side of AVM is shown in Figure 4.

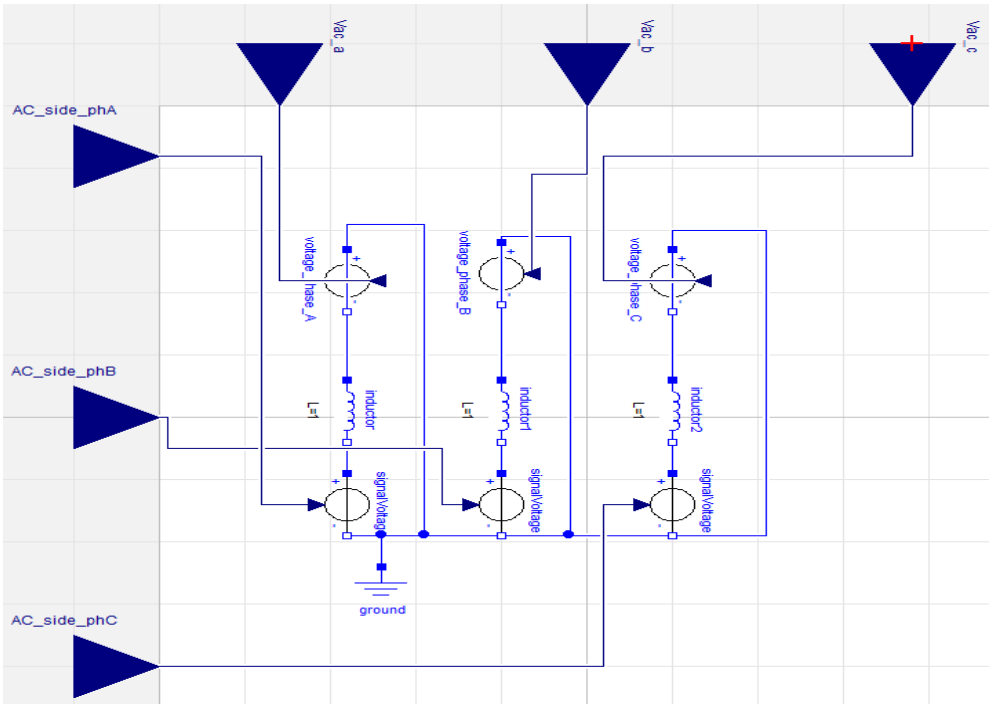

Figure 4 Modelica implementation of the AC-side of AVM

3.1.2. DC side representation of the AVM

Equation (16) represents the relation between the input AC voltage and the output DC voltage of the AVM model.

$$v_{convj} = v_{ref_j} \frac{V_{dc}}{2} \tag{16}$$

where v_{ref_j} are the reference voltages generated by the inner controller of the Upper level controller, which is an input to the AVM. Using the power balance principle the DC-side can be derived into the power equation (17) and current equations (18)

$$V_{dc} I_{dc} = \sum_{j=a,b,c} v_{convj} i_j \tag{17}$$

$$I_{dc} = \frac{1}{2} \sum_{j=a,b,c} v_{refj} i_j \tag{18}$$

The equivalent capacitor shown in Figure 5 is also derived using the energy conservation principle and can be expressed as:

$$C_e = \frac{6C}{N} \tag{19}$$

The model of the DC-side (see Figure 5) also includes an equivalent inductance to substitute the original topology of the VSC-MMC model. This equivalent inductance is given by equation (20):

$$L_{arm_{dc}} = \left(\frac{2}{3}\right) L_{arm} \tag{20}$$

A resistor is used to model the total conduction losses from the VSC-MMC topology, which follows the definition in equation (21):

$$R_{loss} = N R_{ON} \tag{21}$$

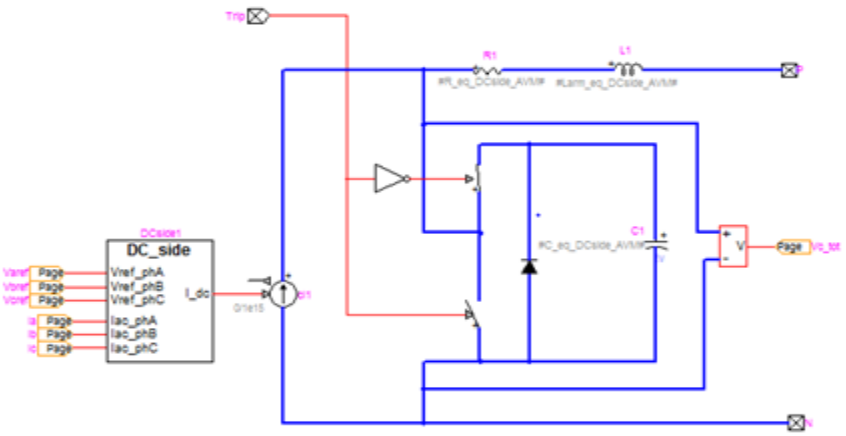

Figure 5 EMTP implementation of DC-side of AVM

Equations (17) and (18) have been implemented in Modelica in order to obtain both direct current and direct voltages. Other components from the MSL have been used to implement the DC-side in Modelica shown in Figure 6.

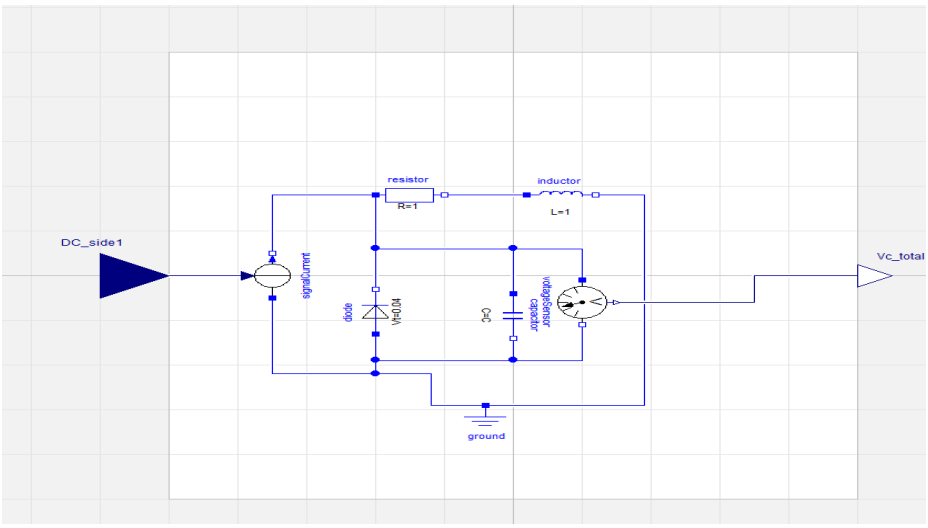

Figure 6 DC side of AVM model in Modelica

3.2. Control System

The principle of VSC-MMC can be understood by a simple two bus system model (see **Error! Reference source not found.**) where V_s is the AC voltage source and V_{conv} is the AC input voltage of the converter. Impedance x symbolizes the equivalent inductances from the transformer between AC source and AC converter.

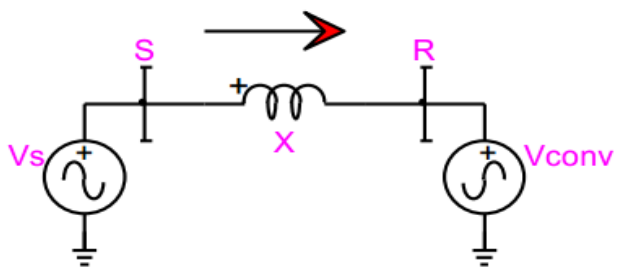

Figure 7 Two bus system representing the functionality of VSC-MMC control system

The concept of power transfer from the source to the converter can be explained by using equation (22) and (23) that represents the amount of active and reactive power transmitting from the source towards the converter that can be expressed as follows:

$$\begin{cases} P_R = \dfrac{V_s V_{conv}}{X} \sin(\delta) & (22) \\ Q_R = \dfrac{V_s V_{conv} \cos(\delta) - V_{conv}^2}{X} & (23) \end{cases}$$

where, δ is the angle between the two voltages . If, angle δ is assumed to keep small then the active and reactive power equations can be linearized as follows [16]:

$$\begin{cases} P_R \simeq \dfrac{V_s V_{conv}}{X} \delta & (24) \\ Q_R \simeq \dfrac{V_{conv}(V_s - V_{conv})}{X} & (25) \end{cases}$$

VSC-MMC control system is composed of three levels of controls; (1) Dispatch control for managing the operation set points; (2) Upper-level control, receives value references for P, Q, Vdc, Vac, and generates the proper signals from synchronized AC generations or from AC limited generation (islands); and (3) Lower-level controls that are responsible for the development of firing pulses necessary to produce the AC waveforms that were requested by the upper level controls. The upper level control is conceived as a high-level control structure, for controlling the optimal transmission of power, voltage and current, independent of the valve topology from the transmission line [11]. For AVM model only upper model control is required and hence in this project the implementation of upper level control has been accomplished.

3.2.1. Upper level control

A schematic representation of the Upper-level control implementation is shown in Figure 8. The Upper-level control generates the three phase reference voltage signals that feed the AVM. This Upper-level control is comprised of different algebraic calculations and controls.

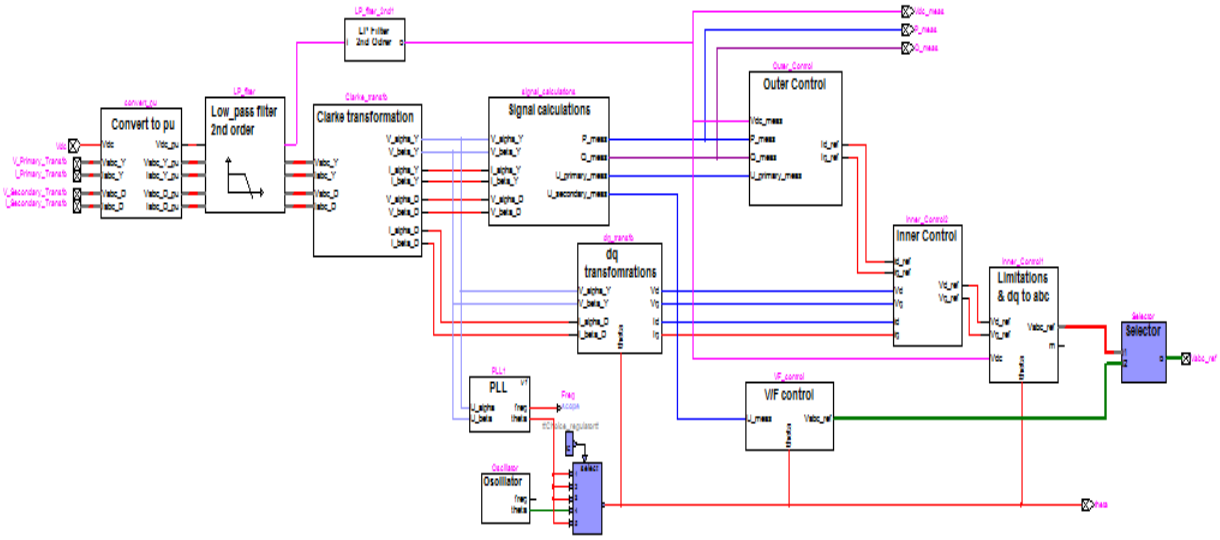

Figure 8 EMTP Implementation of the Upper level Control

All the signals entering into the Upper-level control block are converted to per unit quantities and then filtered by using a Low-pass filter having cut off frequency of 2 kHz. Upper-level control block has two important functional blocks inside which are: (1) power-angle control represented by **V/F control** and vector-current control represented by the (2) **Outer Control** and (3) **Inner Control** blocks. The behavior of the (2) and (3) control blocks is controlled by PID control strategy, only using the proportional and integral gain, with an anti-windup function. This anti-windup function prevents the integral parts from accumulating errors when the output value reaches to its predefined set limit and thus control performance is improved. The integral and proportional gains of each PI controller are computed automatically based on the setting time which can be modified by the user. Outputs from both control systems can be used as reference voltages for AVM model. The required quantities for the control blocks are obtained from the 'Clarke transformation', 'Signal calculations' and 'dq transformations' blocks. These three different controls and the additional blocks used to implement these systems will be described in more details in the following sections.

3.2.1.1 V/F control

Generation of three phase AC voltage requires three variables which are magnitude, phase angle and frequency. V/F control uses the angle and the frequency generated by the PLL or the oscillator block. The AC voltage magnitude Δv_{grid} is controlled by the PID controller.

$$\Delta v_{grid} = \left(v_{ref} - U\_meas\right) \tag{26}$$

The reference implementation of this control is shown in Figure 9. The variable V_{mag} is the output control voltage of the PID controller. This voltage is converted into the three phase output reference voltage of the controller and leveled as V_{abc_ref}.

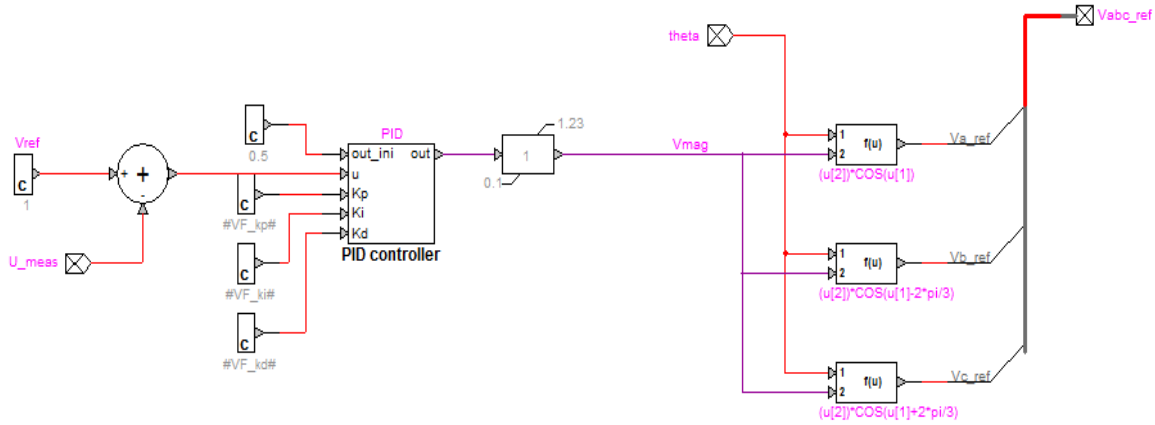

Figure 9 EMTP implementation of V/F control

Equations (27), (28) and (29) describe the mathematical model of the V/F control block. These equations have been implemented in Modelica, with the combination of additional blocks from the MSL. Input parameter theta (θ) is obtained from the PLL or the Oscillator block.

$$V_a\_ref = V_{mag} Cos\theta \tag{27}$$

$$V_b\_ref = V_{mag} Cos(\theta - \frac{2\pi}{3}) \tag{28}$$

$$V_c\_ref = V_{mag} Cos(\theta + \frac{2\pi}{3}) \tag{29}$$

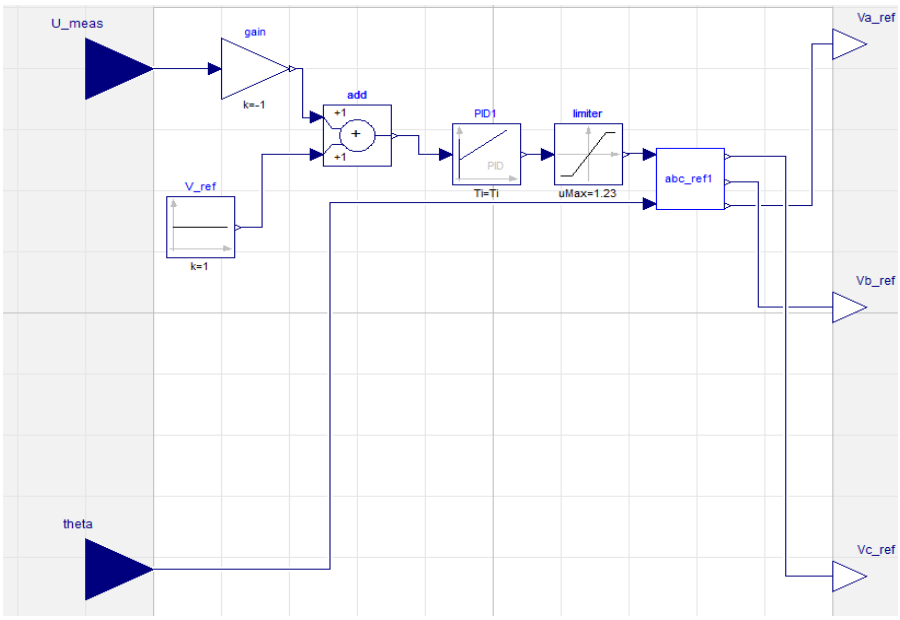

Figure 10 Modelica implementation of V/F control

3.2.1.2 Vector-current control

Outer control and Inner control blocks together make the vector-current control. The primary objective of the vector current control is to be able to control the instantaneous active and reactive power independently by a fast inner control loop. This control system has the advantage of limiting the current that flows into the converter during any disturbance. The Modelica implementation of this control system required a careful study of the inner blocks and the equations that have been used to implement this control system in EMTP. To better understand how the implementation is achieved, in this section blocks and equations have been studied separately. Afterwards, the implementation of a single block has been done and adapted to Modelica.

3.2.1.3 Outer control

Outer controller provides the inner current with necessary reference currents (Id_ref , Iq_ref) (see Figure 11). The outer control can be configured to regulate the outputs in different ways: (1) regulating active power (P control), (2) r(lating the DC voltage (V_{dc} control), (3) droop control (P/V_{dc} control), (4) regulating the reactive power (Q control) and (5) regulate the AC voltage (V_{ac} control).

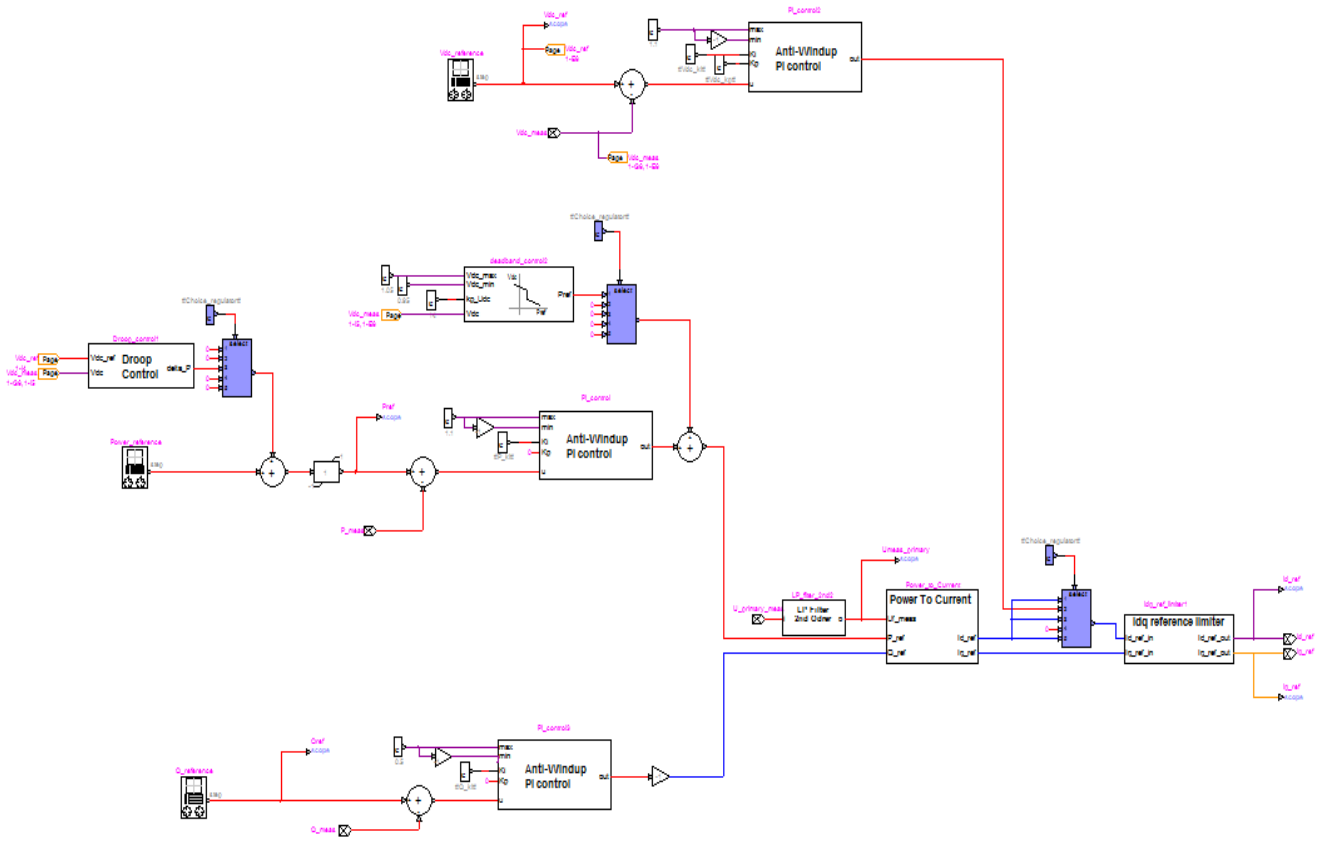

Figure 11 EMTP implementation of Outer control

I. Active power control

The alignment of the grid voltage vector is with the d axis. That makes the q component of the grid voltage equal to zero and d component becomes equal to the voltage magnitude. Thus the active power becomes:

$$P = v_d i_d \tag{30}$$

An integral control produces the necessary d current reference that is Id_ref. Behavior of P control is defined by the following equation:

$$i_{d_{ref}} = \frac{1}{v_d}\left(k_p + \frac{k_i}{s}\right)\left(P_{ref} - P\right) \tag{31}$$

II. DC voltage control (Vdc control)

From the MMC-AVM model, the SM capacitors can be represented as an equivalent capacitor C_{dc}. As the energy in the equivalent inductor L_{DC} is small it can be neglected and the following equation is obtained:

$$C_{dc}\frac{dV_{dc}}{dt} = i_d - I_{dc} \tag{32}$$

If the feed-forward component I_{dc} is neglected and a PI controller is applied in order to regulate the DC voltage then it becomes:

$$i_{d_{ref}} = \left(k_p + \frac{k_i}{s} \right) \left(V_{dc_{ref}} - V_{dc} \right) \tag{33}$$

III. P/Vdc Droop control

The basic concept of droop-control in the DC grid is same as the droop-control in the AC grid. In the AC grid the relationship between frequency and active power is utilized while in the DC grid voltage is a function of active power and droop coefficient which can be expressed as:

$$k_{droop} = \frac{\Delta V_{dc}}{\Delta p} \tag{34}$$

IV. Reactive Power control

Since the grid voltage vector is aligned with the d-axis the q component of the grid voltage is equal to zero and d component equal to the voltage magnitude [16]. The equation becomes:

$$Q = -v_d i_q \tag{35}$$

An integral control is used in order to generate the desired q current reference (iq_ref). The Q control is goverened by the following equation:

$$i_{q_{ref}} = -\frac{1}{v_d} \left(\frac{k_i}{s} \right) \left(Q_{ref} - Q \right) \tag{36}$$

The Modelica implementation of this vector-current control system has not been straightforward. EMTP software implements the PID controller following the mathematical equation where proportional, integral and derivative gains are used:

$$MV(\mathrm{t}) = \mathrm{K}_p \left(e(t) + K_i \int_0^t e(\tau)\, d\tau + K_d \frac{d}{dt} e(t) \right) \tag{37}$$

where, MV is the manipulated variable, K_p is the proportional gain, K_i is the integral gain and K_d is the derivative gain. Modelica implementation of the PID control is available on the MSL, but it uses the time domain equation for the controller, replacing the proportional gain with integral time and derivative time:

$$MV(\mathrm{t}) = \mathrm{K}_p \left(e(t) + \frac{1}{T_i} \int_0^t e(\tau)\, d\tau + T_d \frac{d}{dt} e(t) \right) \tag{38}$$

where, T_i is the integral time and T_d is the derivative time. So, additional calculation out of the Modelica implementation has been done to properly set up the parameters of the PID controller in Modelica in order to emulate the behavior given by the EMTP implementation:

$$K_i = \frac{K_p}{T_i} \text{ and } K_d = K_p T_d \tag{39}$$

Figure 12 shows the Modelica implementation of Outer control block

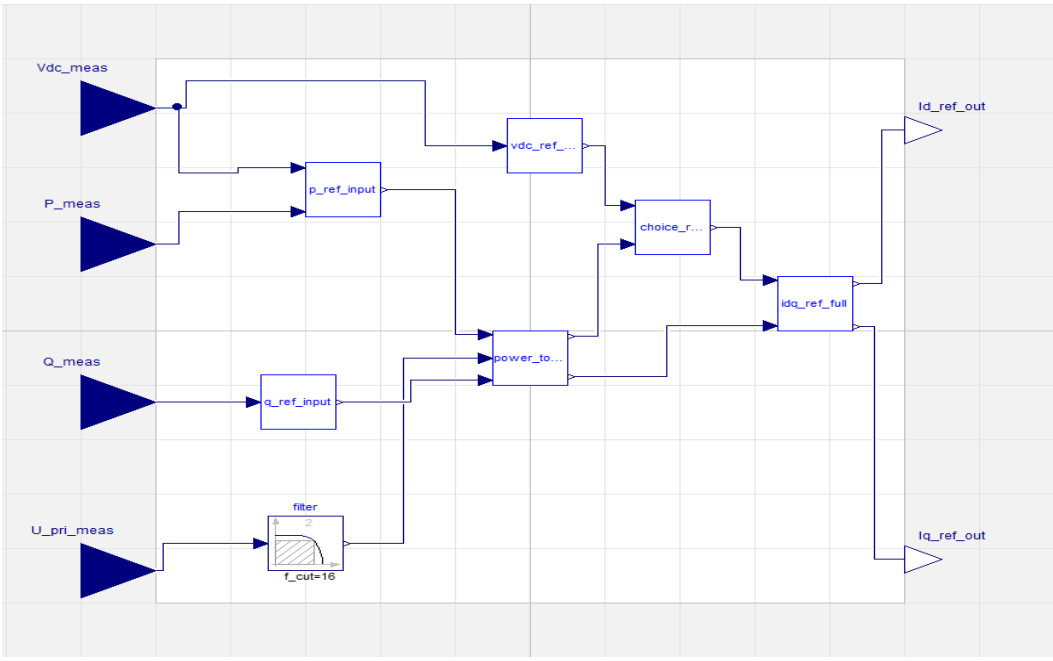

Figure 12 Modelica implementation of Outer control

3.2.1.4 Inner control

This control system performs the decoupling of the of the d and q components of current flow where the grid voltage is used as the phase reference. Inner control block controls the reference voltages Vd_ref and Vq_ref . The behavior of the inner control block is governed by the following equations:

$$v_{conv_d} = -\left(i_{refd} - i_d\right)\left(k_p + \frac{k_i}{s}\right) + v_d + \left(\frac{L_{arm}}{2} + L_{transfo}\right)\omega i_q \tag{40}$$

$$v_{conv_q} = -\left(i_{refq} - i_q\right)\left(k_p + \frac{k_i}{s}\right) + v_q - \left(\frac{L_{arm}}{2} + L_{transfo}\right)\omega i_d \tag{41}$$

Figure 13 and Figure 14 show the EMTP and Modelica implementation of the Inner control block, respectively.

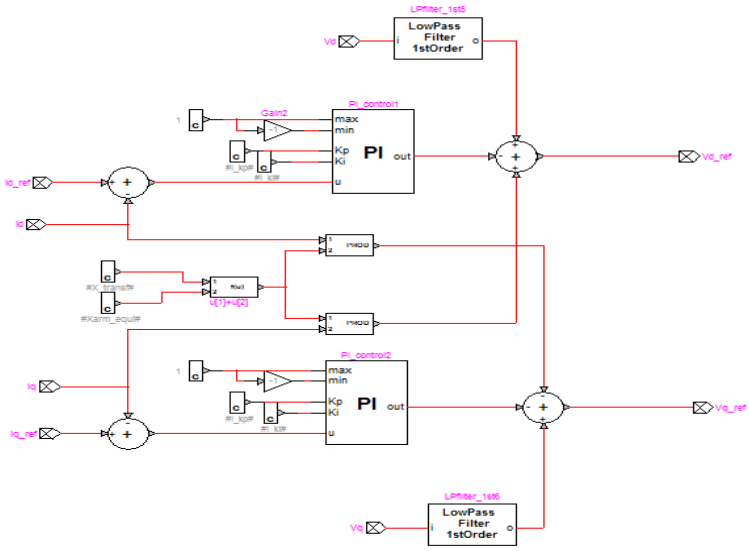

Figure 13 EMTP implementation of Inner control

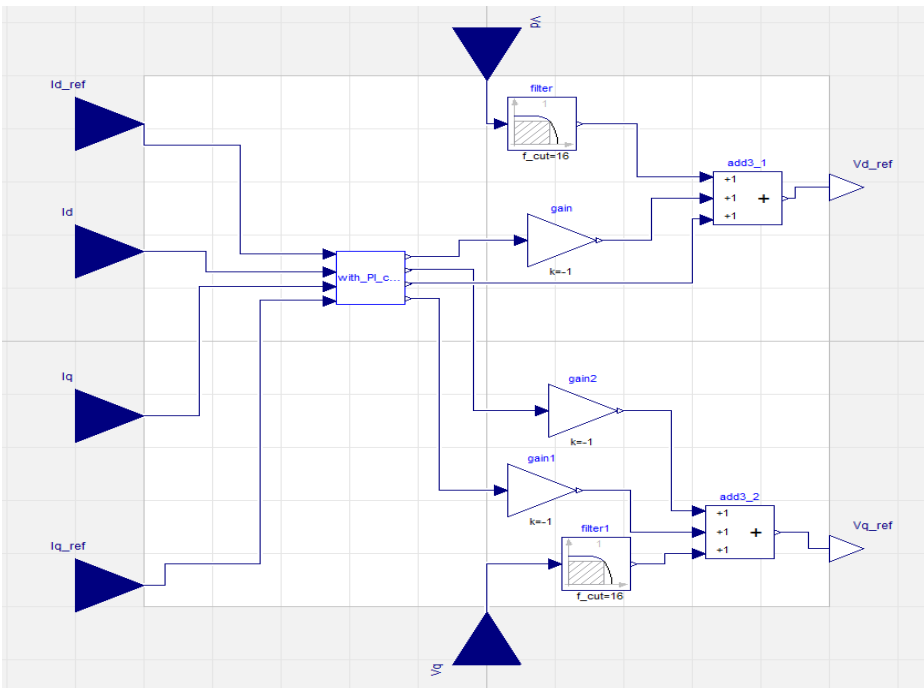

Figure 14 Modelica implementation of Inner control

3.3. Clarke transformation

Changing of variable is a way to introduce simplicity in calculating differential equations. Using Clarke transformation a stationary circuit is transformed to a stationary reference frame. In this transformation a third variable known as 'zero sequence component' is introduced to make the transformation invertible. The Clarke transformation block is composed of four blocks inside for applying Clark transformation to current and voltage signals. The blocks are named as Clarke_Y1, Clarke_Y2, Clarke_D1 and Clarke_D2.

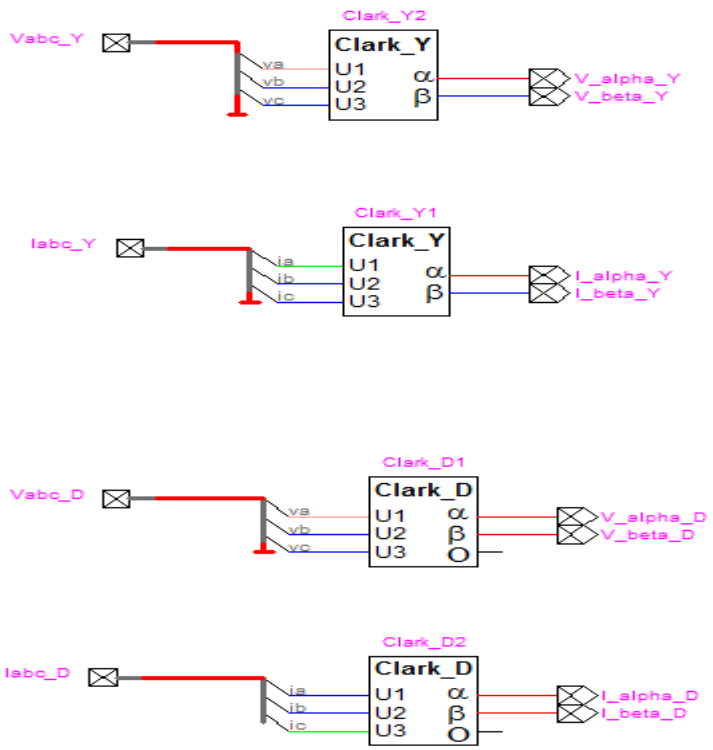

Figure 15 EMTP implementation of Clarke_transfo

Signal input for Clarke_Y1 block is a three phase current ($Iabc\_Y$). The application of the Clarke transformation results in two output quantities I_alpha_Y and I_beta_Y . In this block, Clarke transformation is implemented with the following equations:

$$I\_alpha\_Y = \frac{1}{\sqrt{3}}(I_a - I_c) \tag{42}$$

$$I\_beta\_Y = -\frac{1}{3}(I_a + I_c) + \frac{2}{3}V_b \tag{43}$$

And equations (42) and (43) have been implemented as follows:

```
model Clarke_Y1

equation
 I_alpha_Y = (2/3)*((sqrt(3))/2* I_a-(sqrt(3))/2* I_c);

 I_beta_Y = (2/3)*(-0.5* I_a+ I_b- 0.5* I_c);

  a;
end Clarke_Y1;
```

Figure 16 Modelica implementation of Clarke _Y1 block

Signal input for Clarke_Y2 block is a three phase voltage ($Vabc\_Y$). The application of the Clarke transformation, results in two output quantities V_alpha_Y and V_beta_Y . In this block, Clarke transformation is implemented with the following equations:

$$V\_alpha\_Y = \frac{1}{\sqrt{3}}(V_a - V_c) \tag{44}$$

$$V\_beta\_Y = -\frac{1}{3}(V_a + V_c) + \frac{2}{3}V_b \tag{45}$$

And equations (44) and (45) have been implemented as follows:

```
model Clarke_Y2

equation
 V_alpha_Y = (2/3)*((sqrt(3))/2* V_a-(sqrt(3))/2* V_c);

 V_beta_Y = (2/3)*(-0.5* V_a+ V_b- 0.5* V_c);

  a;
end Clarke_Y2;
```

Figure 17: Modelica implementation of Clarke _Y2 block

Signal input for Clarke_D1 block is a three phase voltage ($Vabc\_D$).The application of the Clarke transformation, results in two output quantities V_alpha_D and V_beta_D . In this block, Clarke transformation is implemented with the following equations:

$$V\_alpha\_D = \frac{2}{3}V_a - \frac{1}{3}(V_b + V_c) \tag{46}$$

$$V\_beta\_D = \frac{1}{\sqrt{3}}(V_b - V_c) \tag{47}$$

And equations (46) and (47) have been implemented as follows:

```
model Clarke_D1

equation

 V_alpha_D = (2/3)*(V_a-0.5*V_b-0.5*V_c);

 V_beta_D = (2/3)*((sqrt(3))/2* V_b-(sqrt(3))/2* V_c);

 zero_sequence_voltage= (1/3)*(V_a+V_b+V_c);

  a;
end Clarke_D1;
```

Figure 18 : Modelica implementation of Clarke _D1 block

Signal input for Clarke_D2 block is a three phase current ($Iabc\_D$). The application of the Clarke transformation, results in two output quantities I_alpha_D and I_beta_D . In this block, Clarke transformation is implemented with the following equations:

$$I\_alpha\_D = \frac{2}{3}I_a - \frac{1}{3}(I_b + I_c) \tag{48}$$

$$I\_beta\_D = \frac{2}{3}(I_b - I_c) \tag{49}$$

And equations (48) and (49) have been implemented as follows:

```
model Clarke_D2

equation
 I_alpha_D = (2/3)*(I_a-0.5*I_b-0.5*I_c);

 I_beta_D = (2/3)*((sqrt(3))/2* I_b-(sqrt(3))/2* I_c);

 zero_sequence_current = (1/3)*(I_a+I_b+I_c);

  a;
end Clarke_D2;
```

Figure 19: Modelica implementation of Clarke _D2 block

3.4. Signal calculations

The purpose of this block is to convert signal inputs from the Clarke transformation block: V_alpha_Y, I_alpha_Y, V_alpha_D and, to four other quantities of P_meas, Q_meas, $U\_primary\_meas$ and $U\_secondary\_meas$ which correspond to the four signals: P, Q, Vdc and Vac that the Upper-level control regulates.

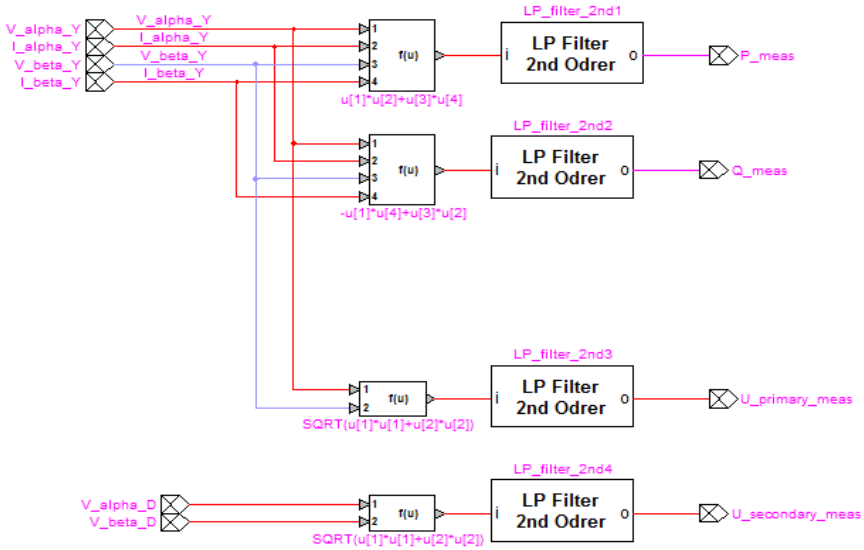

Figure 20 EMTP implementation of Signal Calculations block

The EMTP implementation of this block implements the following equations:

$$P\_meas = (V\_alpha\_Y * I\_alpha\_Y) + (V\_beta\_Y * I\_beta\_Y) \tag{50}$$

$$Q\_meas = (-V\_alpha\_Y * I\_beta\_Y) + (I\_alpha\_Y * V\_beta\_Y) \tag{51}$$

$$U\_primary\_meas = \sqrt{(V\_alpha\_Y)^2 + ((V\_beta\_Y)^2} \tag{52}$$

$$U\_secondary\_meas = \sqrt{(V\_alpha\_D)^2 + ((V\_beta\_D)^2} \tag{53}$$

To obtain the equivalent model in Modelica, previous equations (50)-(53) have been implemented (see Figure 21).

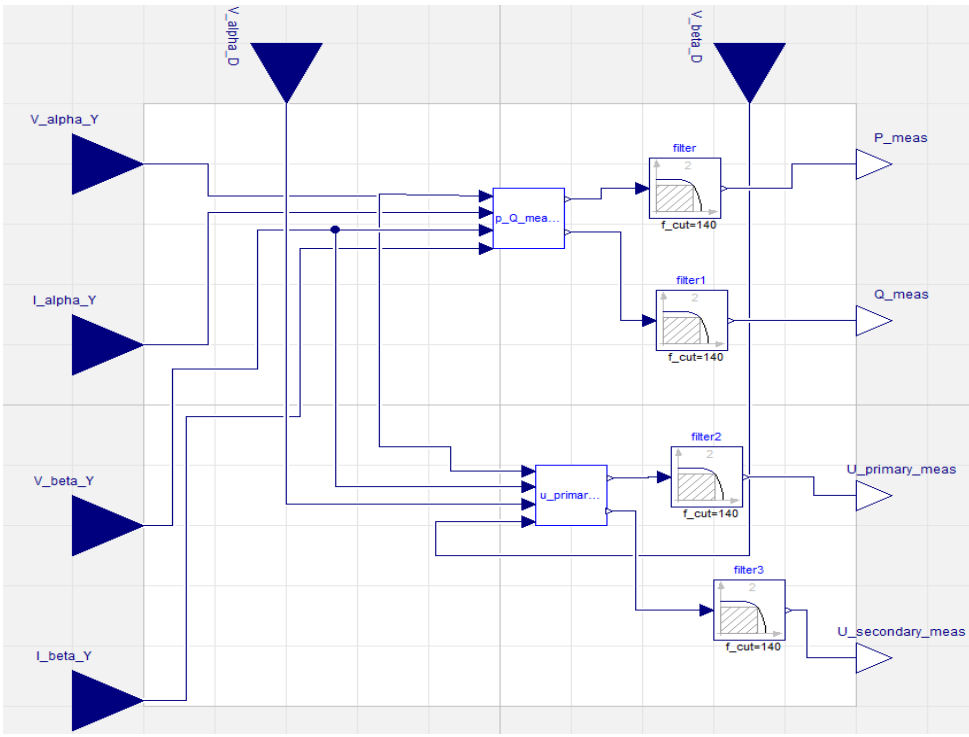

Figure 21 Modelica implementation of Signal calculations block

3.5. dq transformation

dq transformation' block resolves the three-phase AC voltages and currents into d and q components. The functionality of this block is modeled by the Transformation matrix T in equation (54).It transforms the three phase voltage and currents into two quadrature axis components that rotates at a synchronous speed of $\omega = \frac{d\theta}{dt}$. The phase angle θ is obtained from the oscillator block or the PLL block that allows the synchronization of the control parameters with the system voltage.

$$T = \frac{2}{3}\begin{bmatrix} \cos(\omega t) & \cos\left(\omega t - \frac{2\pi}{3}\right) & \cos\left(\omega t + \frac{2\pi}{3}\right) \\ -\sin(\omega t) & -\sin\left(\omega t - \frac{2\pi}{3}\right) & -\sin\left(\omega t + \frac{2\pi}{3}\right) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \tag{54}$$

In matrix T, the direct axis d is aligned with the grid voltage. The dq voltage and currents are obtained by using the transformation matrix T and can be expressed as:

$$i_{dq} = T i_{abc} \tag{55}$$

$$v_{dq} = T v_{abc\_grid} \tag{56}$$

Active power, reactive power and the AC grid voltages are calculated from the dq references and can be expressed as:

$$P = v_d i_d + v_q i_q \tag{57}$$

$$Q = -v_d i_d + v_q i_q \tag{58}$$

$$v_{grid} = \sqrt{v_d^2 + v_q^2} \tag{57}$$

Inside the dq transformation block there are two blocks named as $Park\_V$ and $Park\_I$, which perform park transformation (see Figure 22). Park transformation is used to eliminate the time variable inductances from the voltage and current quantities.

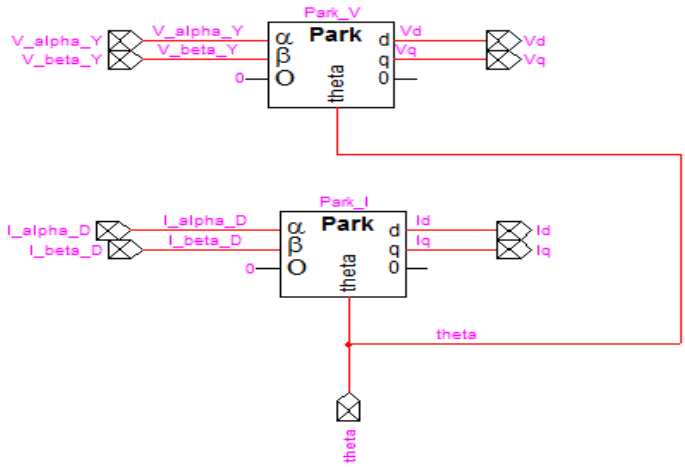

Figure 22 EMTP implementation of dq transformations

This block uses four inputs named as V_alpha_Y, V_beta_Y, I_alpha_D and I_beta_D from the Clarke transformation block and $theta$ from the PLL or Oscillator block .

$Park\_V$ block converts the voltage quantities V_alpha_Y and V_beta_Y into voltages quantities of Vd and Vq .

$$V_d = V_a \, Sin\,\theta - V_b Cos\theta \tag{59}$$

$$V_q = V_a Cos\theta + V_b Sin\theta \tag{60}$$

$Park\_I$ Block converts the current quantities of I_alpha_D and I_beta_D into Id and Iq .

$$I_d = I_a \, Sin\,\theta - I_b Cos\theta \tag{61}$$

$$I_q = I_a Cos\theta + I_b Sin\theta \tag{62}$$

Equations from Park_V and Park_I blocks have been implemented in Modelica (see Figure 23 and Figure 24).

```
model park_V

equation

  Vd = Va*sin(theta)-Vb*cos(theta);
  Vq = Va*cos(theta)+Vb*sin(theta);

  a;
end park_V;
```

Figure 23: Modelica implementation of Park _V block

```
model park_I

equation

  Id = Ia*sin(theta)-Ib*cos(theta);
  Iq = Ia*cos(theta)+Ib*sin(theta);

  a;
end park_I;
```

Figure 24 Modelica implementation of Park _I block

Using these Park_V and Park_I blocks, the corresponding Modelica model for the dq transformation is in Figure 25.

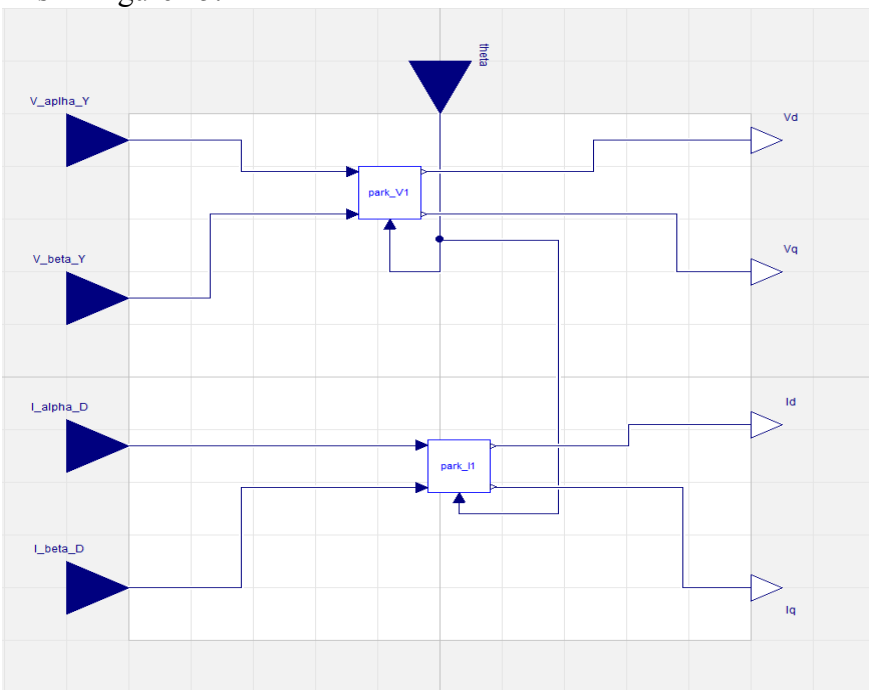

Figure 25 Modelica implementation of dq transformation

3.6. PLL and Oscillator

The main task of the PLL and the oscillator is to generate a phase angle to synchronize with the phase angle and the frequency of the AC grid voltage.

3.6.1. Oscillator

Experiences in simulating the oscillator block in EMTP give an output of sawTooth wave for a constant input of frequency (i.e. 50 Hz in this case).

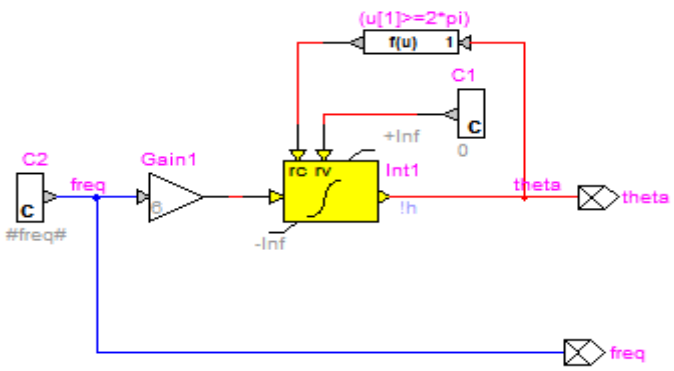

Figure 26: EMTP implementation of Oscillator block [9]

So, this oscillator block has been directly replaced by a sawTooth wave source available in the MSL:

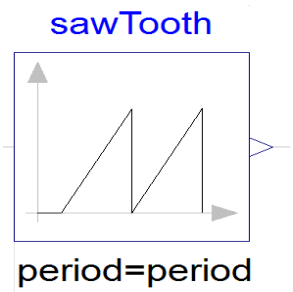

Figure 27 sawTooth wave source in Modelica

3.6.2. PLL

Phase locked loop (PLL) has a variety of uses in various communication, control, automation, and instrumentation systems where there is a need of signal synchronization. In power systems it is largely used for power quality monitoring purposes in case of power system disturbances and to compute reference signals for the internal control loops in uninterruptible power supplies [19].

In this upper control block it has been used for detecting the fundamental phase angle and frequency of the grid. PLL block is comprised of many small blocks in EMTP. All the individual EMTP equivalent blocks have been implemented in Modelica to obtain the functionality of the PLL.

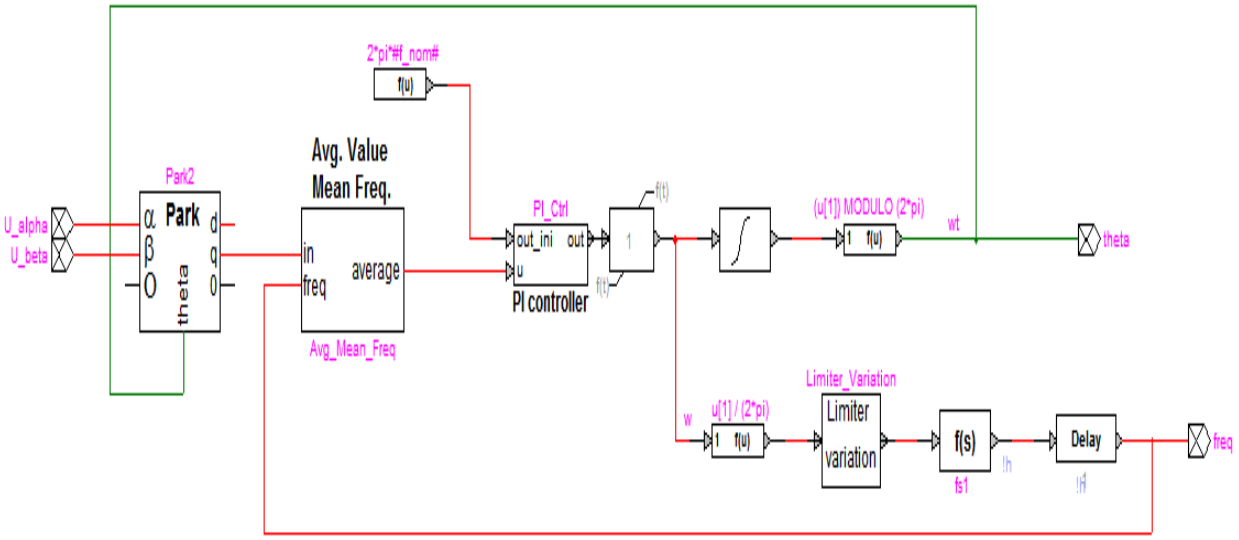

Figure 28 EMTP implementation of PLL block

PLL uses an Avg.ValueMeanFreq block that has been shown in Figure 29.

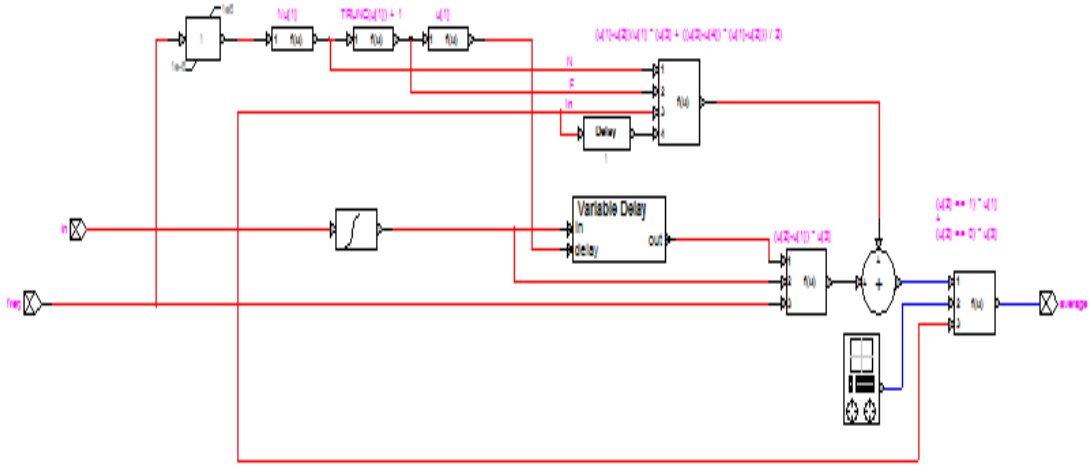

Figure 29 EMTP implementation of Avg.Value Mean.Freq block

In the Avg. ValueMeanFreq presence of the parameter 'diet' indicates the simulation time step of EMTP that is used to simulate the model. For the Modelica implementation, parameter 'dt' is not considered. Instead, the user defines the time step and simulation time directly in simulation options in Modelica. The block also uses a Truncation block that has been implemented in Modelica.

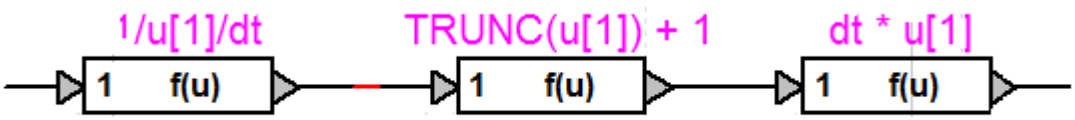

Figure 30 shows a truncation functions with division and multiplication operation where the simulation time step dt is used. This are part of the Avg.ValueMeanFreq block

Figure 33 shows the Modelica implementation of the truncation block.

```
model truncation

equation
  tran_out = integer(trancaton) + 1;

  a;
end truncation;
```

Figure 31: Modelica implementation of Truncation block

Modelica implementation of Avg.ValueMeanFreq block is shown in Figure 32:

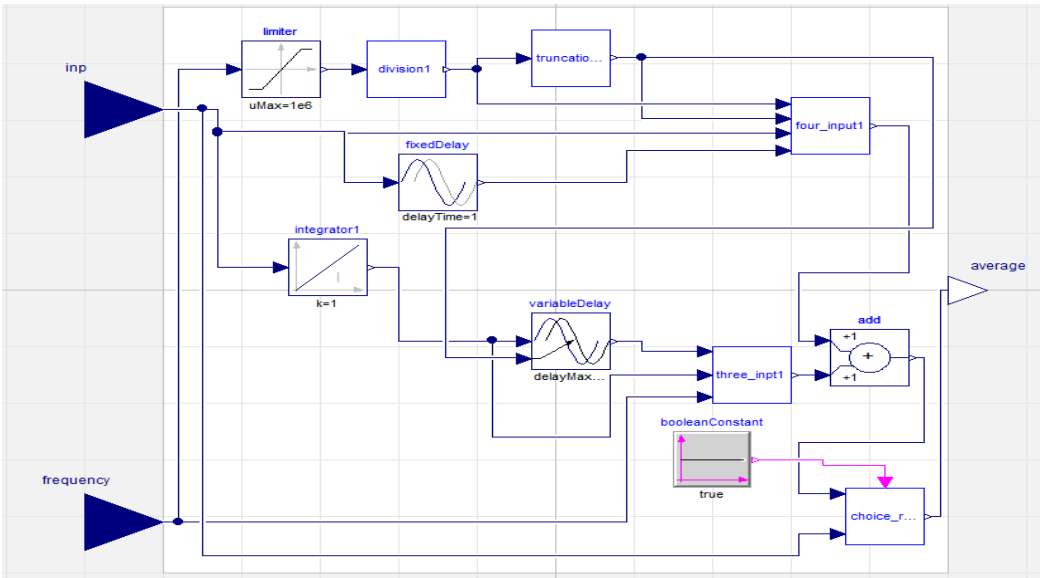

Figure 33 Modelica implementation of Avg.Value Mean.Freq block

PLL block contains a Modulo operation that has been implemented in Modelica shown in Figure 33:

```
model modulo_operation

constant Real pi= Modelica.Constants.pi;
Real p;

equation
  p=2*pi;
  y= mod(u,p);

end modulo_operation;
```

Figure 34: Modelica implementation of Modulo operation

The MSL provides a 'fixedDelay' and 'variableDelay' blocks which have been used for time shifting of signal quantities and can be considered as equivalent of the delay and variable time delay in EMTP.

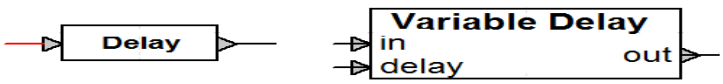

Figure 35 EMTP blocks for time shifting of signals

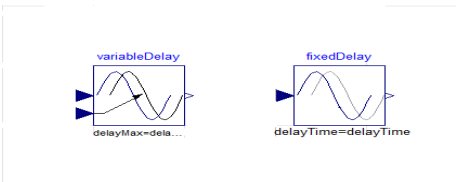

Figure 36 Modelica blocks for time shifting of signals

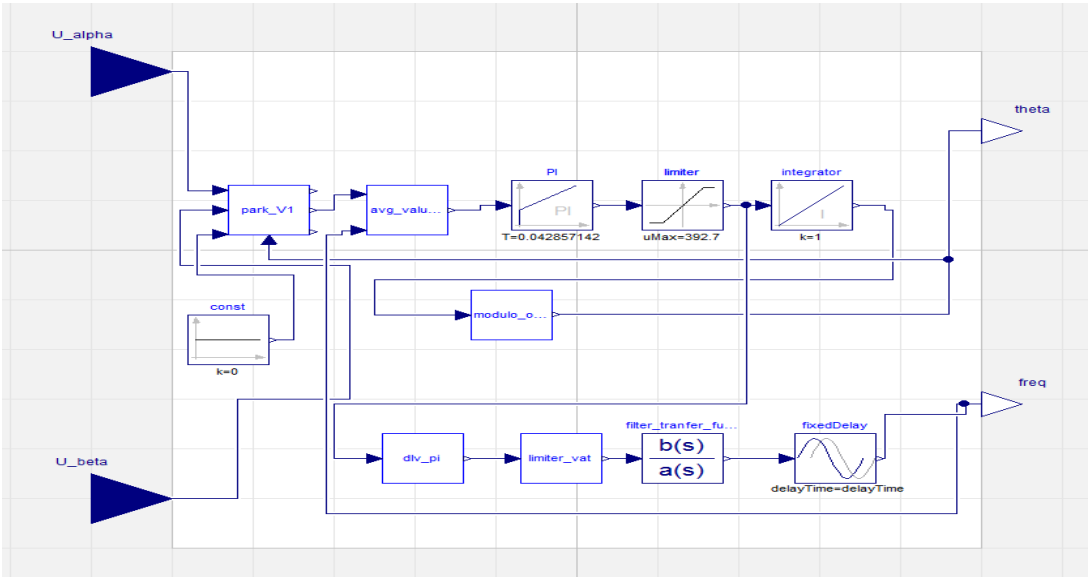

Figure 37 Modelica implementation of PLL

3.7. Limitations & dq to abc

Limitations & dq to abc block uses the output quantities from the control blocks and convert them to the final output signals of the reference voltage quantities. Then the voltage quantities are converted back from the dq reference frame to the abc reference frame (see Figure 45)

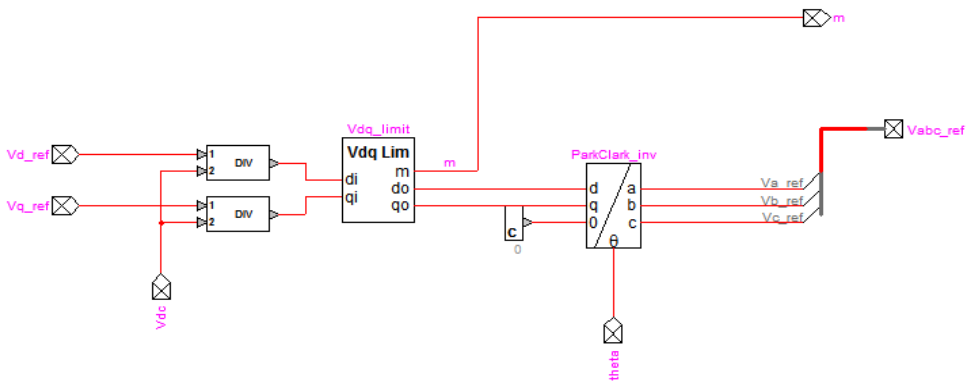

Figure 38 EMTP implementation of Limitations & dq to abc

Vd_ref, Vq_ref and Vdc are the three inputs of the Vdq_limit. The division blocks converts the Vd_ref, Vq_ref and Vdc into two quantities named di and qi. In Vdq_limit block the quantities di and qi are converted from Cartesian to Polar coordinate and the magnitude m (see Figure 38) is limited to 1.5.

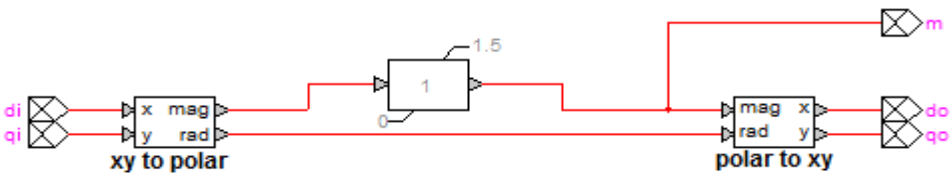

Figure 39 EMTP implementation of vdq limit block

Then the polar values are again converted back from Polar to Cartesian coordinate. Following equations represent the transformation from Cartesian coordinates into Polar coordinates

$$mag = \sqrt{di^2 + qi^2} \tag{63}$$

$$rad = tan^{-1}\left(\frac{qi}{di}\right) \tag{64}$$

And following equations perform the conversion back to Cartesian coordinates from Polar coordinates:

$$do = mag * Cos(rad) \tag{65}$$

$$qo = mag\ sin(rad) \tag{66}$$

Figure 40 and Figure 41 show the Modelica implementation of equations of (63)-(66).

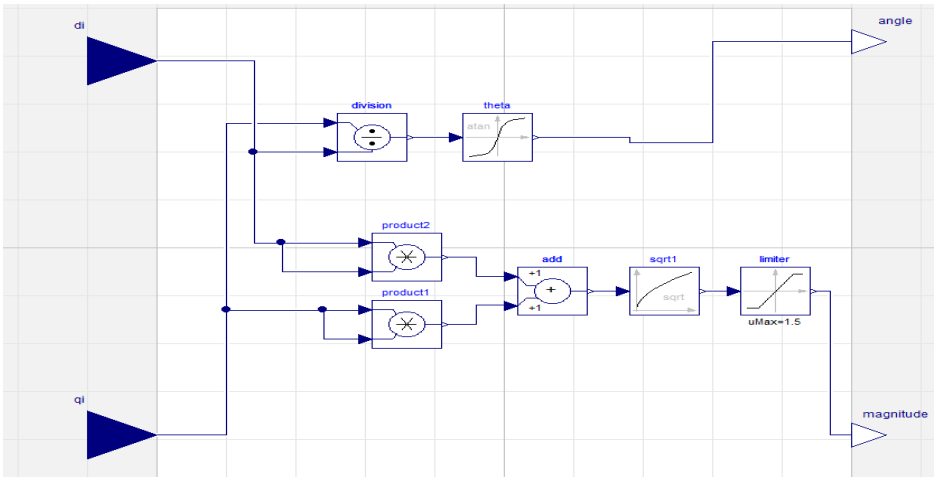

Figure 42 Modelica implementation of xy to polar block

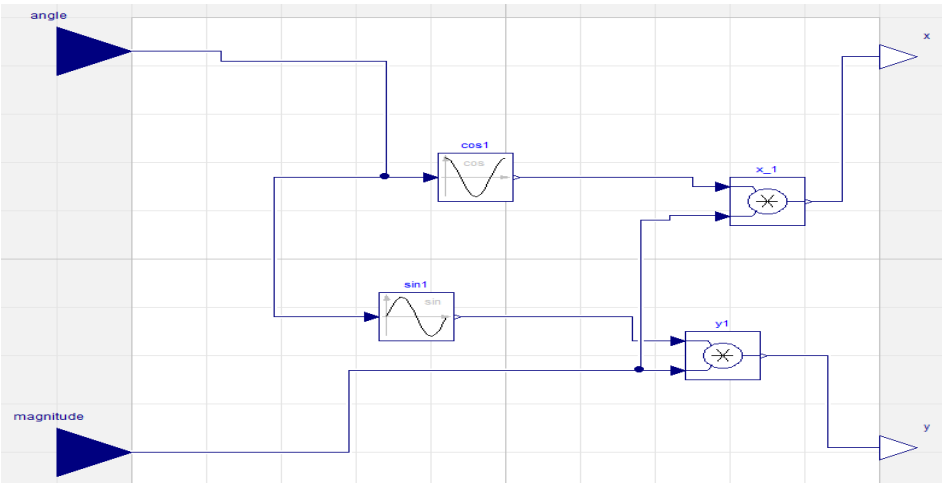

Figure 43 Modelica implementation of xy to polar block

Inside the ParkClark_inv block there are two blocks named as $Parc\_inv2$ and $Clark\_inv1$. In the $Parc\_inv2$ block following equations have been implemented In EMTP to obtain the inverse of the park transformation:

$$V\_alpha = V_d Sin\theta + V_q Cos\theta \tag{67}$$

$$V\_beta = -V_d Cos\theta + V_q Sin\theta \tag{68}$$

$$V\_o = 1 * V\_o \tag{69}$$

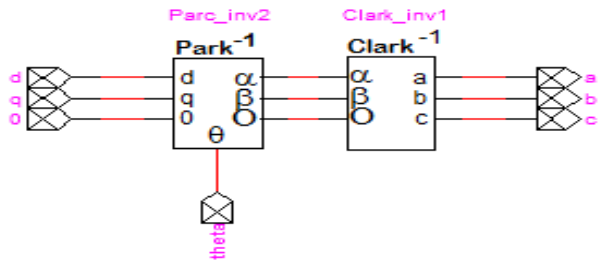

Figure 44 EMTP implementation of ParkClark inverse block

The outputs of the park inverse block are the inputs to the $Clark\_inv1$ block. To perform the inverse Clark transformation following equations has been implemented in Modelica:

$$V\_a\_ref = V\_alpha - V\_o \tag{69}$$

$$V\_b\_ref = \frac{1}{2}(-V\_alpha) + \frac{\sqrt{3}}{2}V\_beta + V\_o \tag{70}$$

$$V\_c\_ref = -\frac{1}{2}V\_alpha - \frac{\sqrt{3}}{2}V\_beta + V\_o \tag{71}$$

Equations (63-71) have been implemented in Modelica to obtain all the individual blocks and then they have been merged together to obtain the 'Limitations & dq to abc' block.

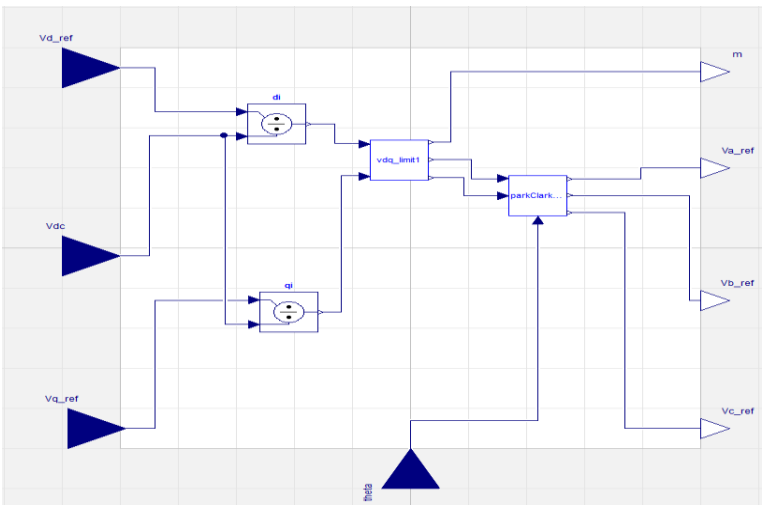

Figure 45 Modelica implementation of Limitations & dq to abc block

Other blocks that perform the operation to transform the quantities suitable for the control blocks are following:

3.8. Convert to pu and Low_Pass filter

In the Convert to pu block shown in Figure 46 all the quantities are converted to per unit quantities with respect to their base values. Per unit quantities are used to have all the measures under the same reference, making calculations easier to perform.

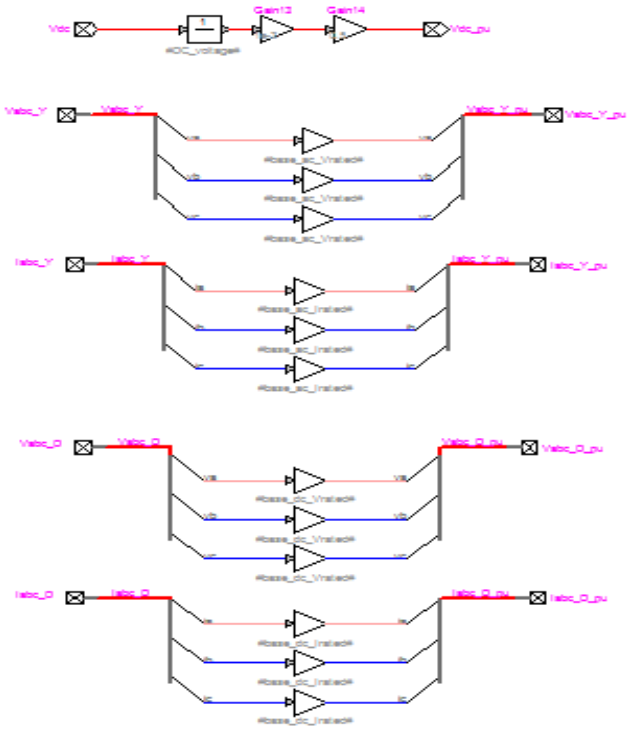

Figure 47 EMTP implementation of Convert to pu block

A very simple implementation, using a gain block, is shown in Figure 48 shows the Modelica for a Vabc_Y_pu block.

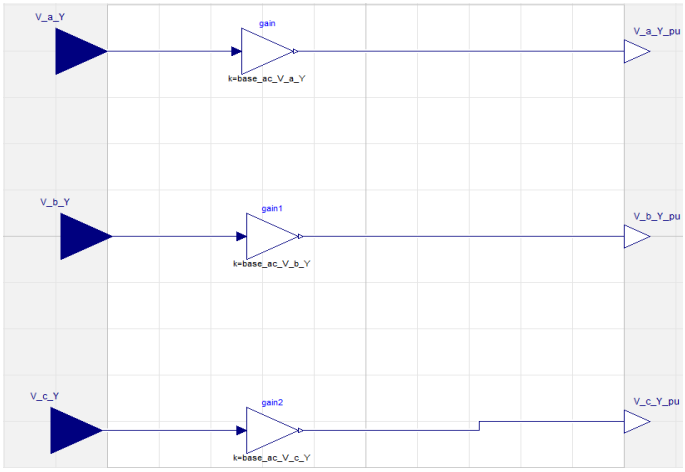

Figure 49 Modelica implementation of Vabc_Y_pu

Likewise, all the other parts of the block have been implemented in Modelica to complete the per unit block. Since the output of the per unit block enters into the filter block, for convenience they have been merged together in Modelica shown in Figure 50.

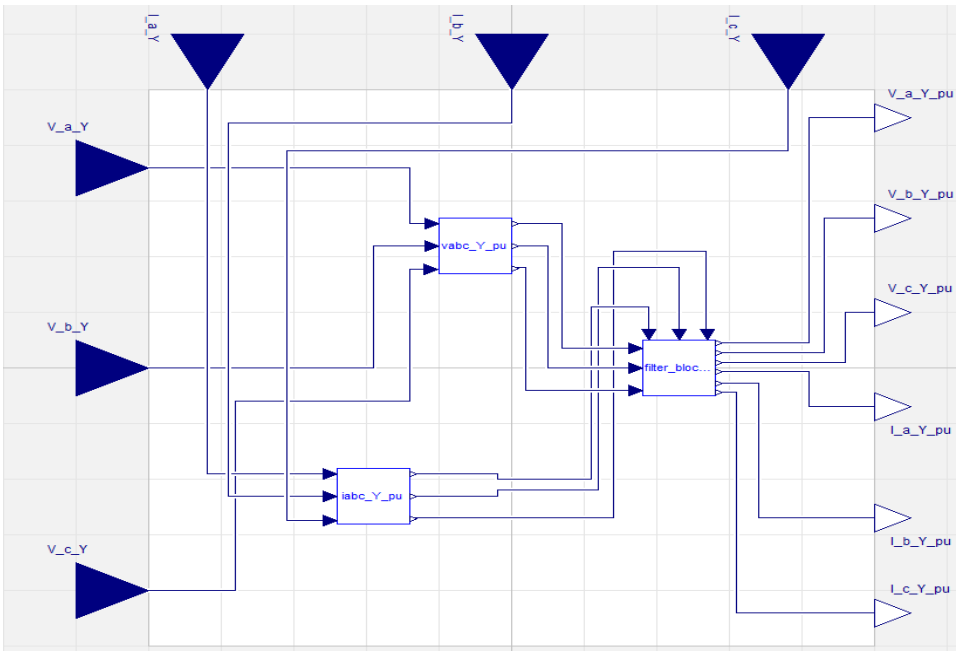

Figure 51 Modelica implementation of Convert to pu and Low_pass filter blocks

3.9. Selector block

The selector block has been designed to choose between the output of the 'Limitations & dq to abc' block and VF_control block.

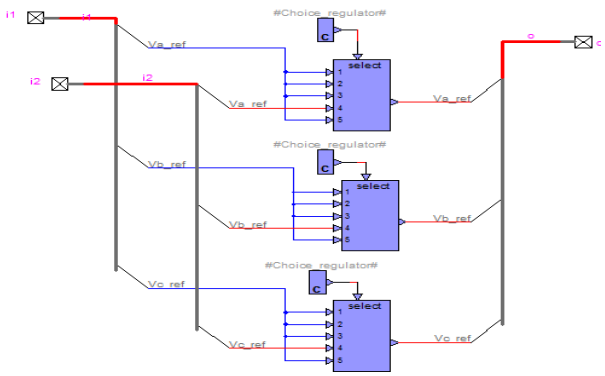

Figure 52 EMTP implementation of Selector block

While a number value is used in the EMTP implementation, it is easy to use a boolean source for selecting options. For a True value of the boolean source the selector selects the output of 'Limitations & dq to abc' block and for a False value of the selector block the output of 'VF_control' block is selected as the output reference voltage.

Figure 53 shows the Modelica implementation of the selector block.

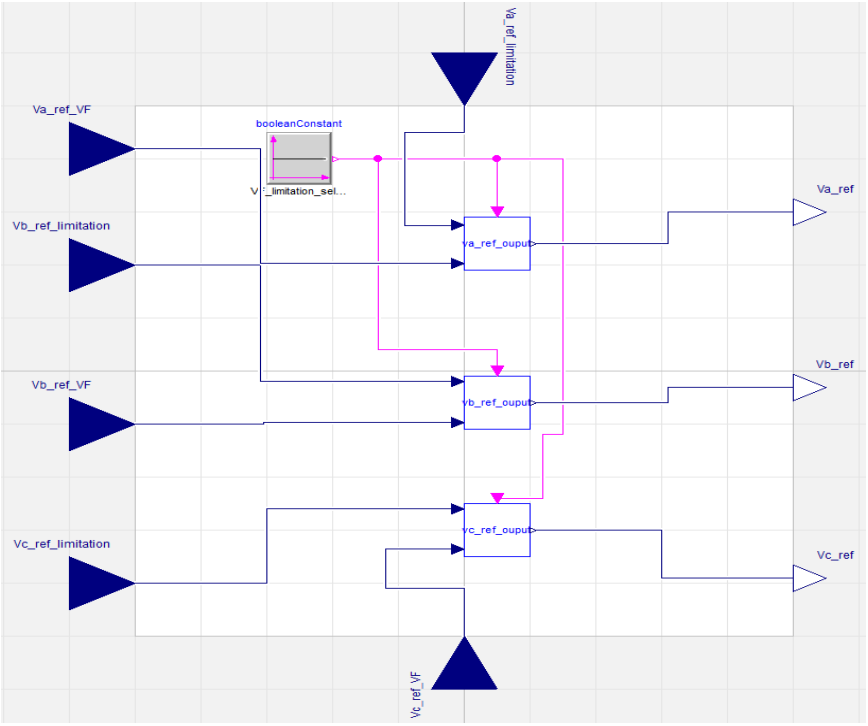

Figure 54 Modelica implementation of Selector block

4. Model Validation

The new components and systems that have been developed in Modelica need to be validated. The process of validation is the last step of the creation of new Modelica components. Every individual Modelica block that has been implemented in Modelica needs to be validated against its equal model from EMTP. In the validation part each block from both the newly developed Modelica model and reference EMTP model have been compared in order to show the similar response when they are applied to the same input. In model validation two of the important considerations are selection of solvers and selection of time step. EMTP uses trapezoidal solver and for Modelica several solvers are available. Out of those 'Dassl' has been selected for model validation.Dassl solver performs the differential algebraic operations using backward differentiation formula and then the resulting nonlinear system is solved by Newton's method. While doing the validation of the models it has been observed that smaller time steps for a larger simulation time gives more accurate results than for the simulation with larger time steps for a shorter time. But if the time steps are too small then EMTP cannot compute the result of a block having a larger number of differential algebraic operations. Again, if the time steps are too large then inaccuracies in the system responses have been observed. So, optimum time steps and simulation time have been chosen for every block in order to obtain comparable results. Each individual block was simulated for an equal amount of time in both EMTP and Modelica. MATLAB has been used to plot the Modelica and EMTP output to compare the results in the same time axis. It is important to choose the EMTP time step in such a way that the total number of intervals should be equal in both Modelica and EMTP to plot the validation result in MATLAB.

4.1. AVM validation test

Sinusoidal AC voltages of the same magnitude and frequency have been used as input to validate the AVM model developed in EMTP and Modelica. Three phase AC voltages with an amplitude of 100 V was chosen in both the cases. Here, phase-a is the reference phase and phase-b and phase-c has a phase shift of 120 and 240 degrees respectively with respect to phase-a. Three phase reference voltages Varef, Vbref and Vcref have been chosen, each of them has an amplitude of 50V. Varef is the reference quantity, and the other two reference voltages are Vbref and Vcref having a phase shift of 120 and 240 degrees with respect to Varef. In the test each inductor has a value of 1 H, every resistor has a value of 1 Ω and every capacitor has a value of 1 F. The capacitor of the DC-side of the AVM was initialized with an initial value of 1500 V across which the output is measured. In Modelica the DC inductor was initialized with an initial current of 1500 A. A time step of 0.005 ms and a simulation time of 0.05 seconds have been chosen in EMTP for the simulation of this circuit. Total number of Modelica intervals for the simulation is 10000.

Figure 55 and Figure 56 shows the AVM test circuits in EMTP and Modelica respectively.

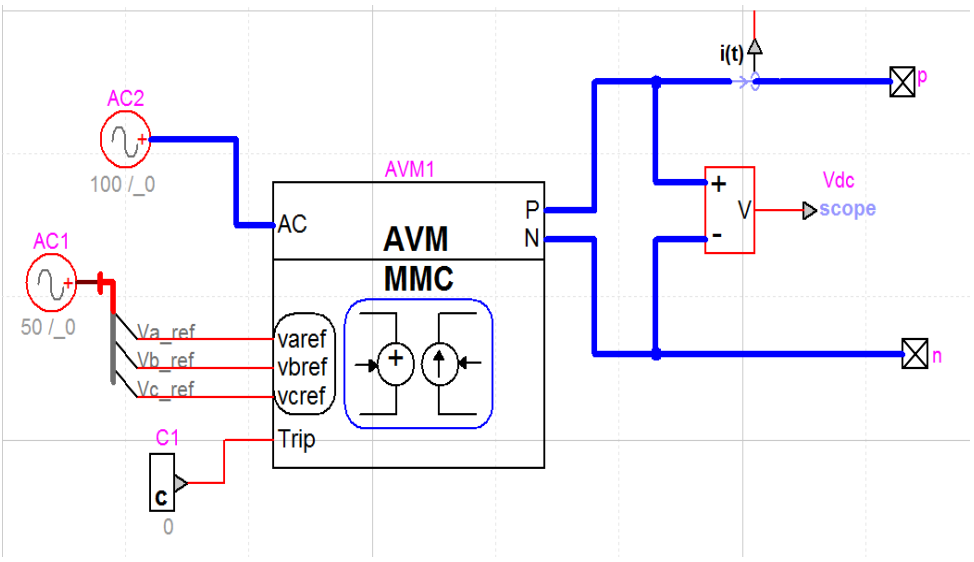

Figure 55 EMTP implementation of AVM test

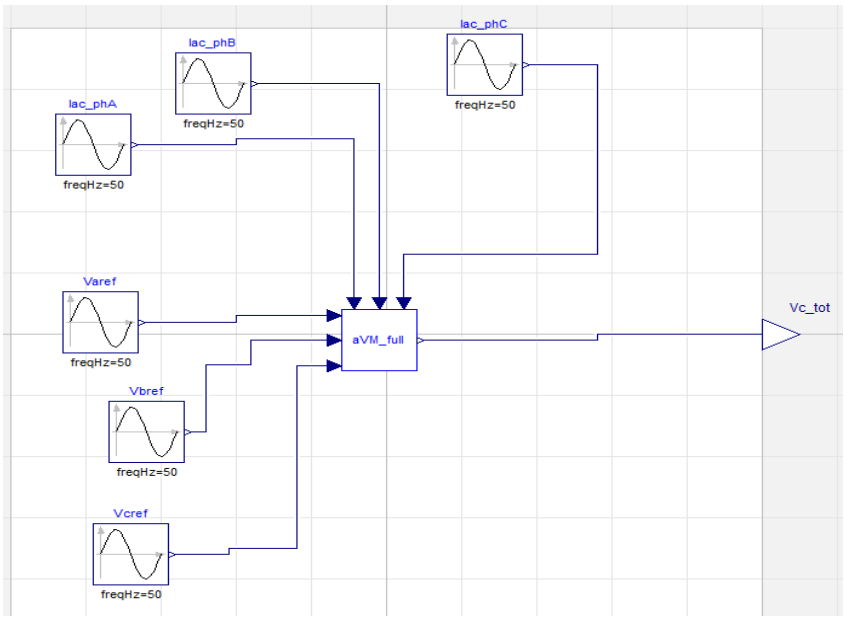

Figure 56 Modelica implementation of AVM test

Figure 57 shows the output response of AVM model both in Modelica and EMTP. The result shows that the Modelica simulation gives an output signal that has the same amplitude of the EMTP simulation. EMTP output, however, presents a rising transient behavior before it shows a steady-state response, which is not present in Modelica response.

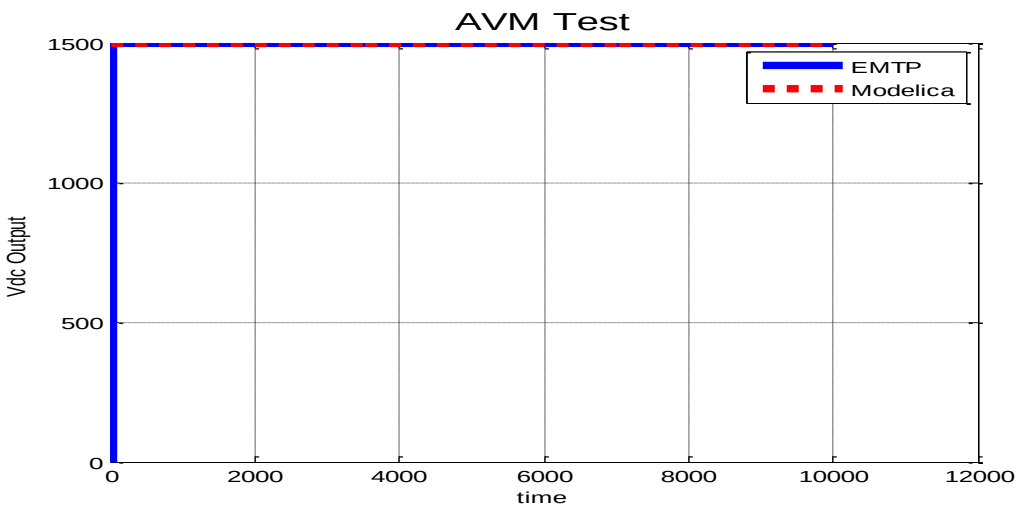

Figure 57 AVM model response for EMTP and Modelica

4.2. Upper control block

Inside the upper control block Clarke transformation, Signal Calculations, dq transformations, PLL, Oscillator and Limitations & dq to abc blocks perform algebraic operations. This is why these blocks have been validated only with steady state input where the input remains constant with respect to time.

4.2.1. Clarke transformation

Three voltage quantities V_a, V_b, V_c and three current quantities are I_a, I_b, I_c are the inputs of the Clarke transformation block. The input quantities are presented in the following table:

Table 1 The input quantities for Clark transformation test

	Clark_Y1	Clark_Y2	Clark_D1	Clark_D2
I_a	6	-	-	4
I_b	5	-	-	6
I_c	3	-	-	8
V_a	-	5	9	-
V_b	-	10	10	-
V_a	-	9	7	-

Figure 51 shows the EMTP test circuits for the Clarke transformation. Figure 52 and 53 respectively show the Modelica test circuits for Y and D quantities of Clarke transformation

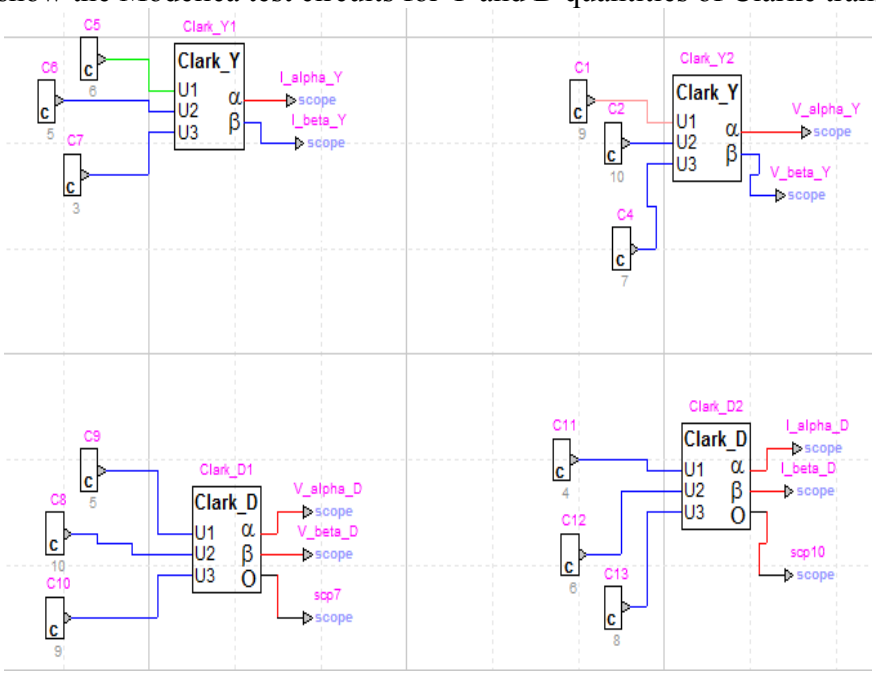

Figure 58 EMTP implementation of test circuit of Clarke transformation

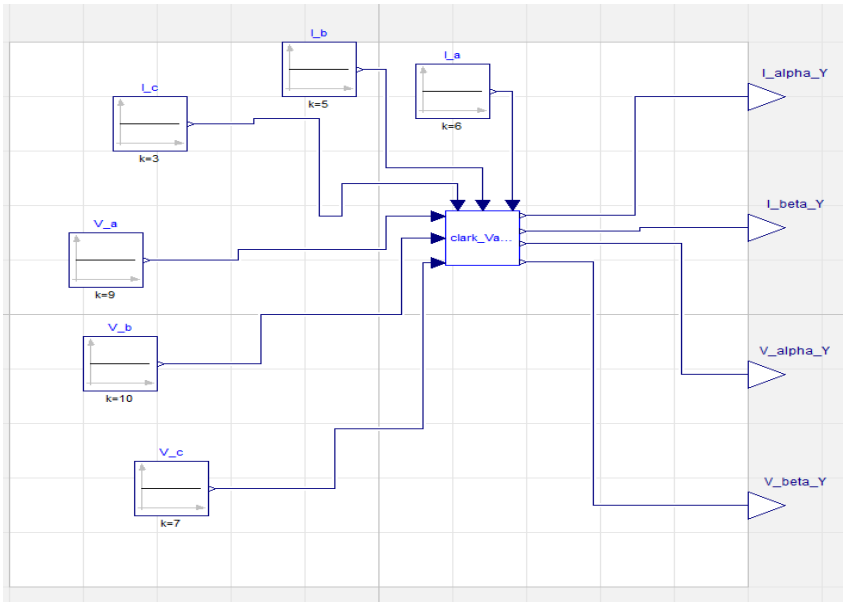

Figure 59 Modelica implementation of Clarke_Y1 and Clarke_Y2

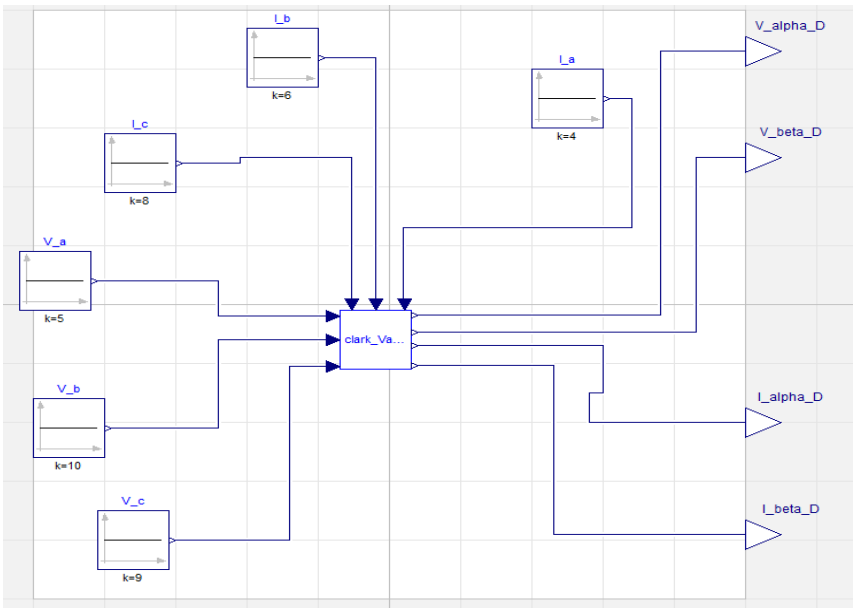

Figure 60 Modelica implementation of Clarke_D1 and Clarke_D2

Figure 61 and Figure 62 respectively show the output response of the Clark transformation in EMTP and Modelica for above mentioned inputs. A time step of 1 ms and a simulation time of 1 second have been chosen in EMTP for the simulation of this circuit. Total number of Modelica intervals for the simulation is 1000. Results obtained from EMTP and Modelica show that the difference is very insignificant. Hence, the Clark transformation block and the Modelica block can be considered as equivalent to one another:

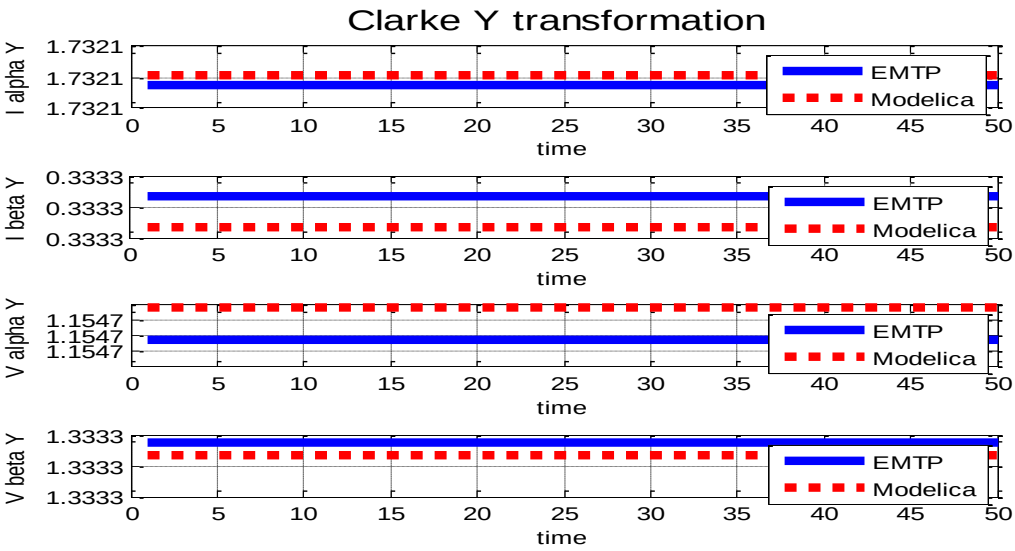

Figure 61 EMTP and Modelica response for Clarke_Y1 and Clarke_Y2 block

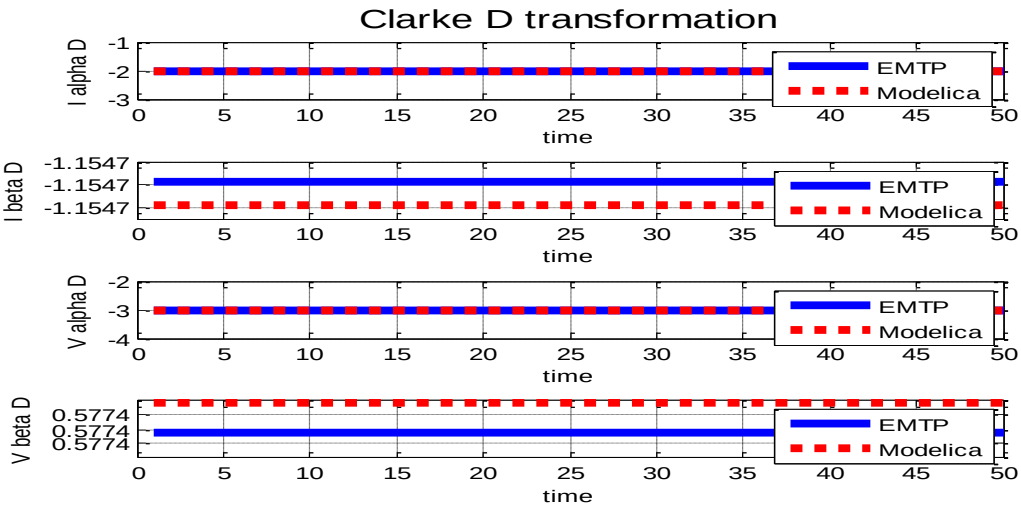

Figure 62 EMTP and Modelica response for Clarke_D1 and Clarke_D2 block

4.2.2. Signal calculations block

Four voltage quantities V_alpha_Y, V_beta_Y, V_alpha_D, V_beta_D and two current quantities I_alpha_Y and I_beta_Y are the input quantities of the signal calculation block. The input quantities are presented in the following table

Table 2 The input quantities for Signal calculation test

V_alpha_Y	5
V_beta_Y	2
V_alpha_D	2
V_beta_D	2
I_alpha_Y	2
I_beta_Y	2

A time step of 1 ms and a simulation time of 1 second have been chosen in EMTP for the simulation of this circuit. Total number of Modelica intervals for the simulation is 1000.Figure 63 and Figure 64 show, respectively, the EMTP and Modelica test circuit for the Signal calculation block.

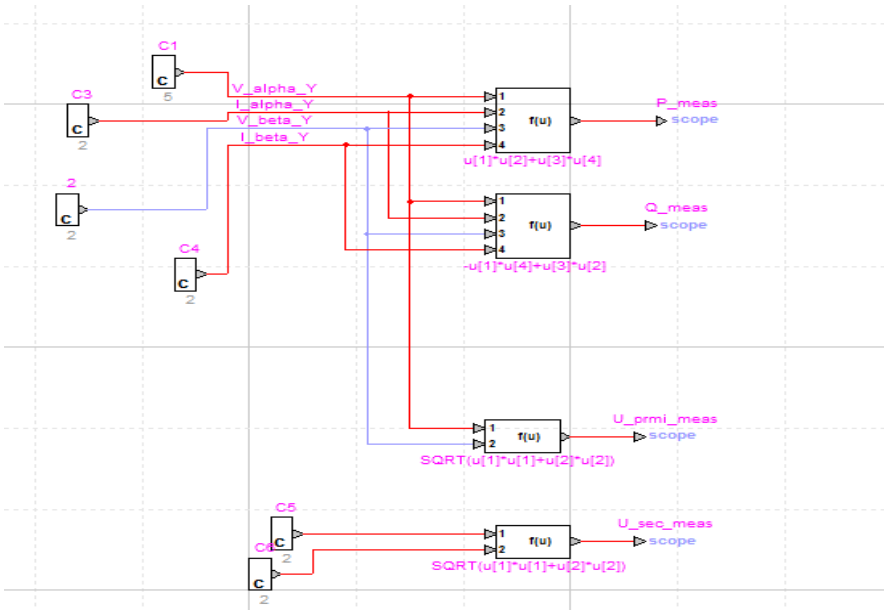

Figure 63 EMTP implementation of the Signal calculation test

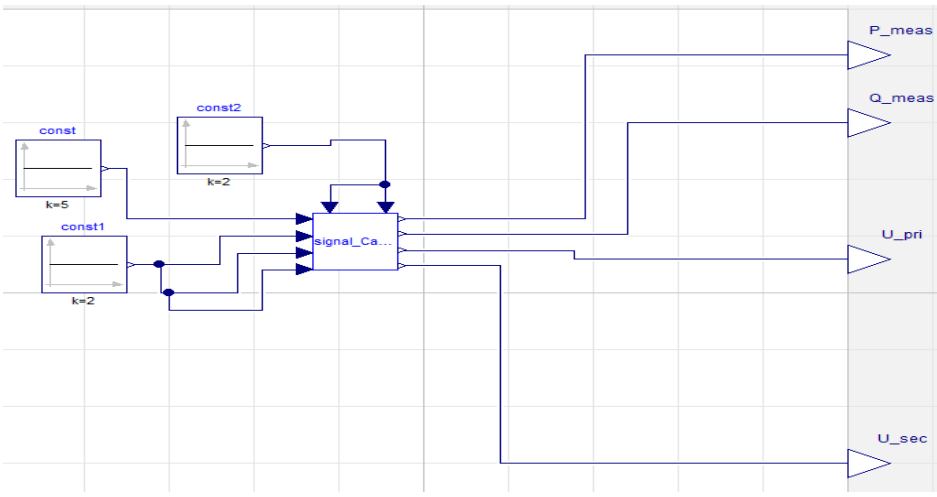

Figure 64 Modelica implementation of the Signal calculations test

Figure 65 shows the output response of signal calculations block for the constant input quantities mentioned above. From the figure it is evident that the Modelica model gives an acceptable equivalent output as the EMTP model.

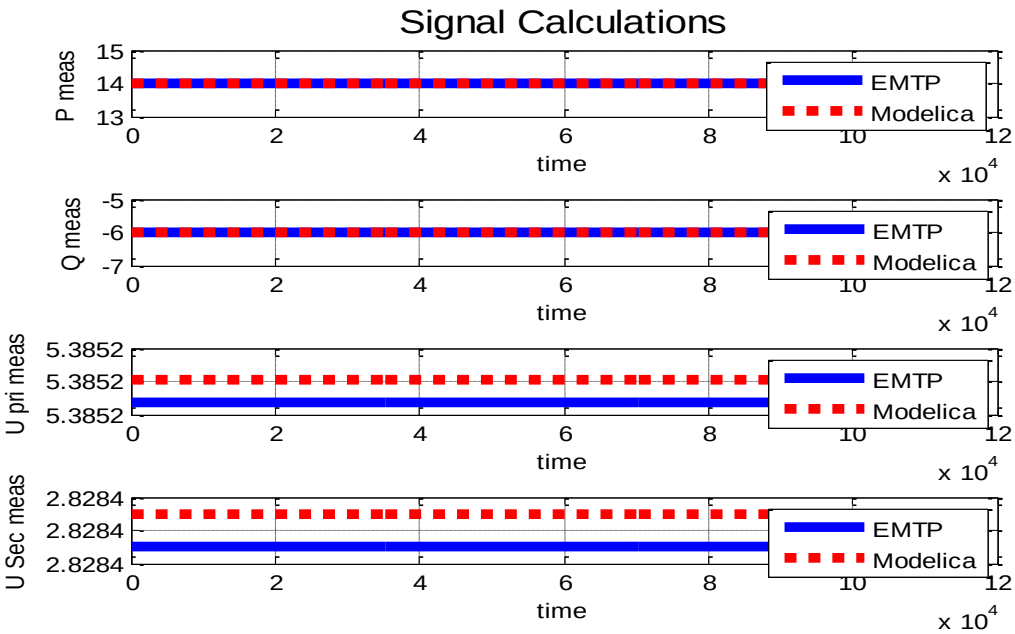

Figure 66 EMTP and Modelica response for Signal calculations block

4.2.3. dq transformations

Two voltage quantities V_alpha_Y, V_beta_Y and two current quantities I_alpha_D and I_beta_D and theta value are the input quantities of the dq transformations block. The input quantities are presented in the following table:

Table 3 The input quantities for dq transformations test

V_alpha_Y	8
V_beta_Y	5
I_alpha_D	2
I_beta_D	2
Theta	32

A time step of 1 ms and a simulation time of 1 second have been chosen in EMTP for the simulation of this circuit. Total number of Modelica intervals for the simulation is 1000.Figure 67 and Figure 68 show the EMTP and Modelica test circuit for dq transformations test.

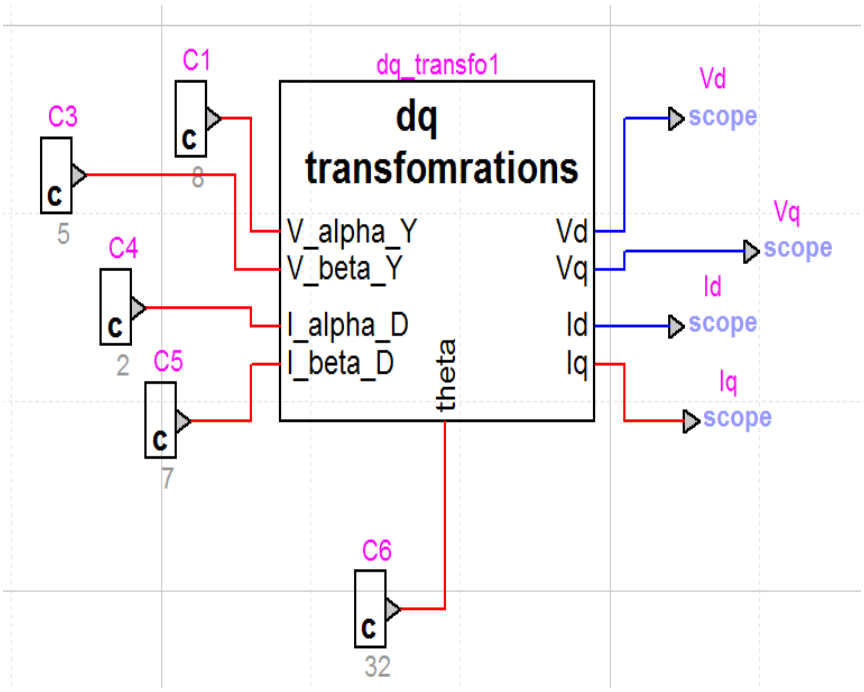

Figure 67 EMTP implementation of the dq transformations test

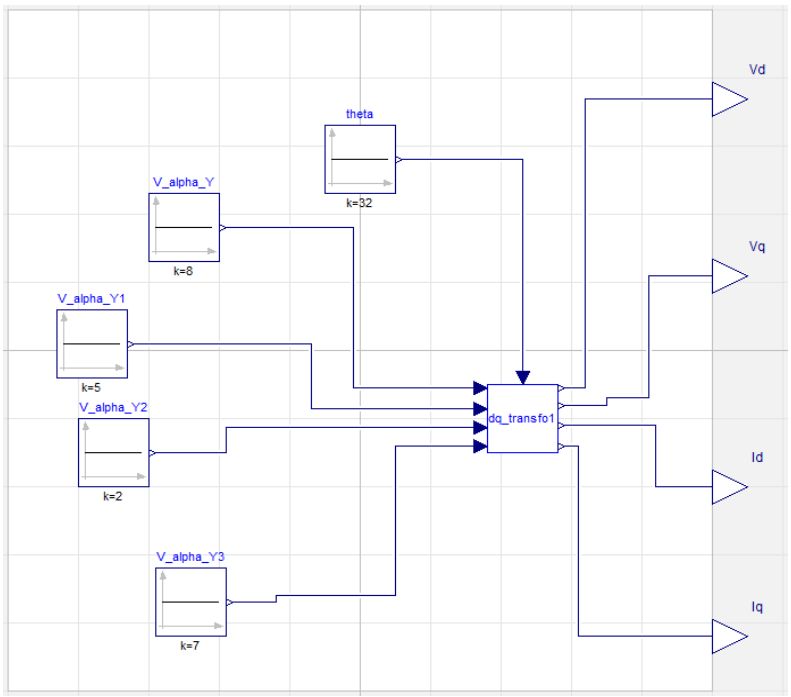

Figure 68 Modelica implementation of the dq transformations test

Figure 69 shows the output response of dq transformations block for the constant input quantities. From the EMTP and Modelica responses it is evident that the Modelica model gives an acceptable equivalent output as the EMTP model.

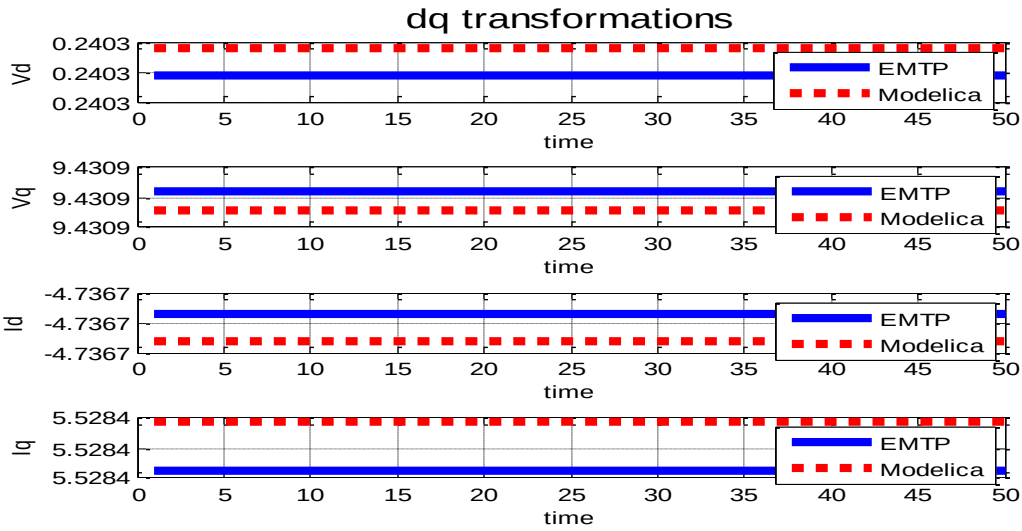

Figure 69 EMTP and Modelica response for dq transformation block

4.2.4. Limitations & dq to abc

Validation of Park^{-1} and Clark-1 block has been presented in Appendix.A.1 and Appendix.A.2. Three voltage quantities Vd_ref , Vd_ref, Vdc and theta are the input quantities of the Limitations & dq to abc block. The input quantities are presented in the following table:

Table 4: The input quantities for dq transformations test

Vd_ref	15
Vd_ref	20
Vdc	5
theta	37

A time step of 1 ms and a simulation time of 1 second have been chosen in EMTP for the simulation of this circuit. Total number of Modelica intervals for the simulation is 1000.Figure 70 and Figure 71 respectively show the EMTP and Modelica test circuit for Limitations & dq to abc block.

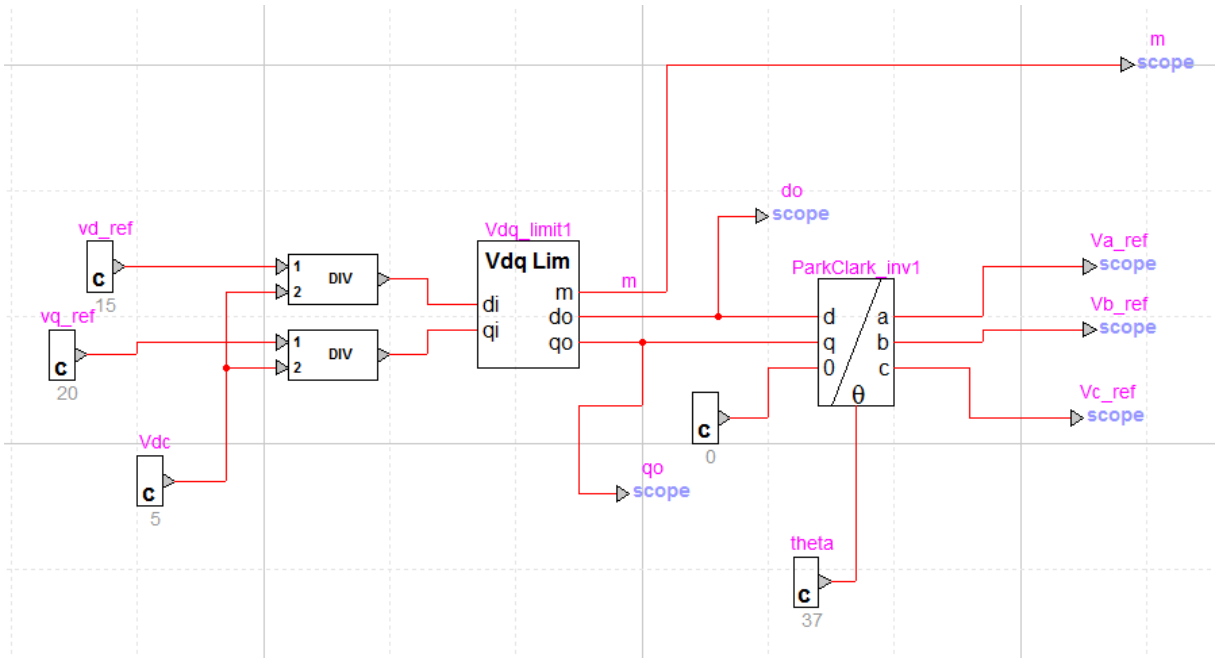

Figure 70 EMTP implementation of the Limitations & dq to abc test circuit

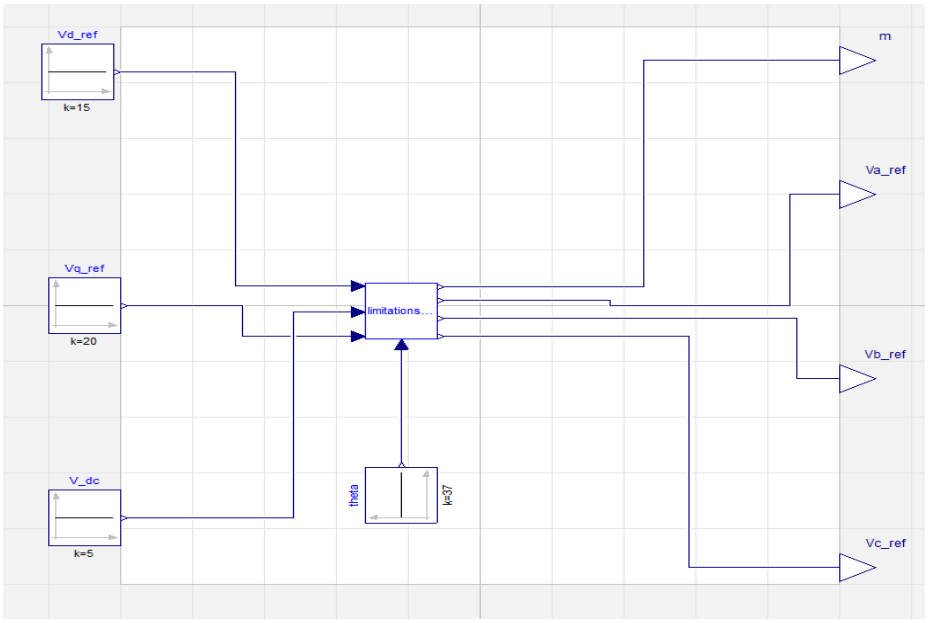

Figure 71 Modelica implementation of the Limitations & dq to abc test circuit

Figure 72 shows the output response of Limitations & dq to abc for constant input quantities mentioned above. From the EMTP and Modelica response it can be concluded that the designed Modelica block and reference EMTP blocks perform the same operation.

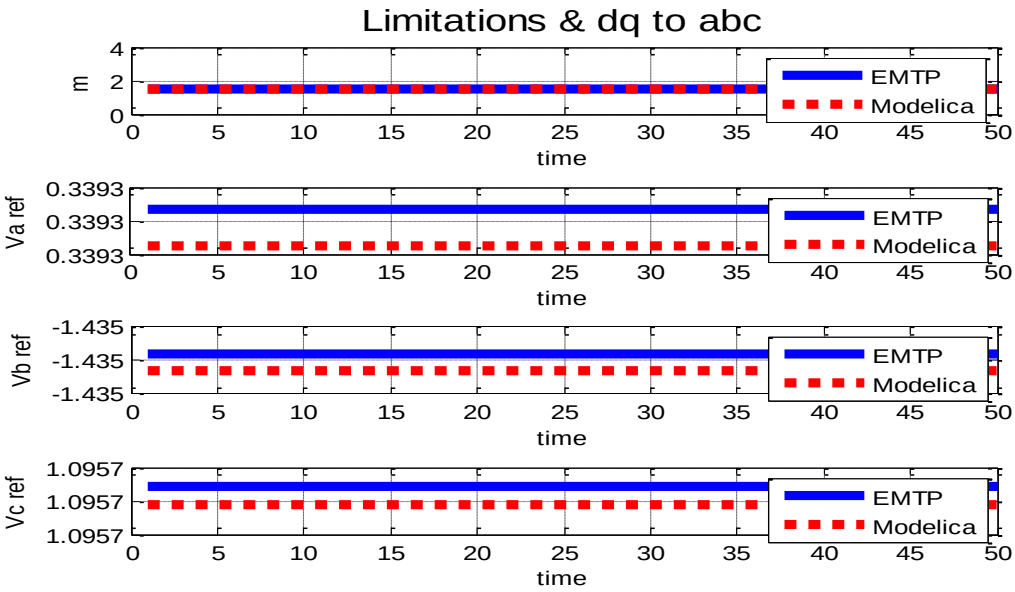

Figure 72 EMTP and Modelica response for Limitations & dq to abc block

4.2.5. Oscillator

As Oscillator block produces output of sawTooth wave for a constant frequency (here 50 Hz) it has been validated with the output of the sawtooth source of the same magnitude and period shown in Figure 73.

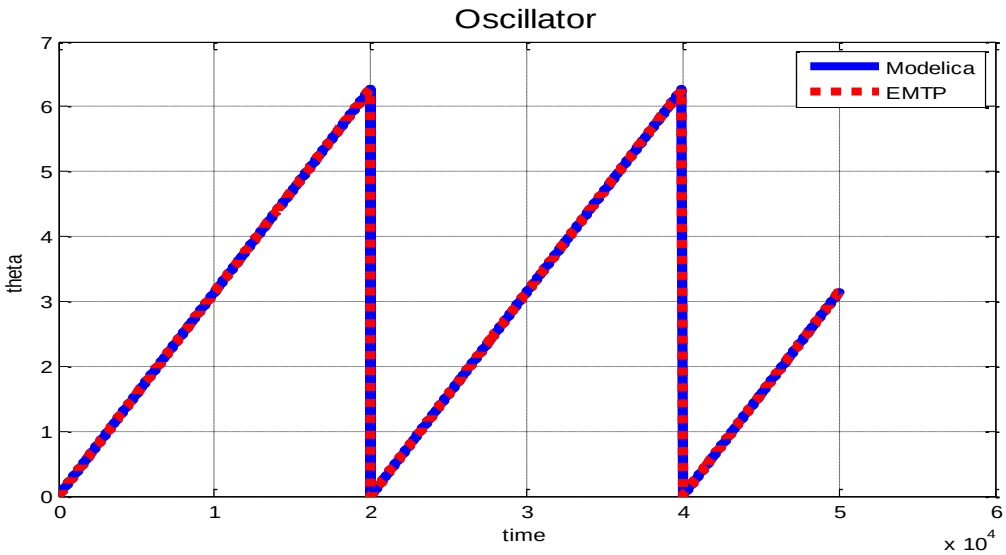

Figure 73 EMTP and Modelica response of Oscillator for a constant frequency input of 50 Hz

4.2.6. PLL

Validation of the Avg.Value Mean.Freq block has been shown in Appendix.B.1. Figure 74 and Figure 75 show the PLL test circuit for for sinus input of U_aplha and cosine input of U_beta for a frequency of 50 Hz where the amplutude of both the Sine and the Cosine input is 10.

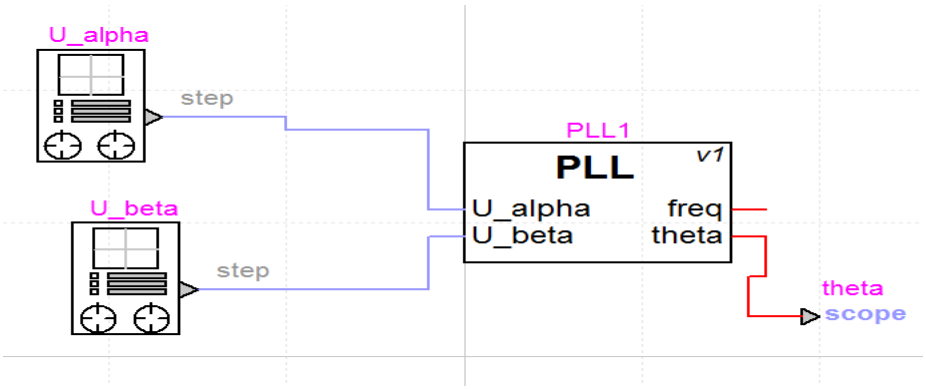

Figure 74 EMTP implementation of PLL test circuit

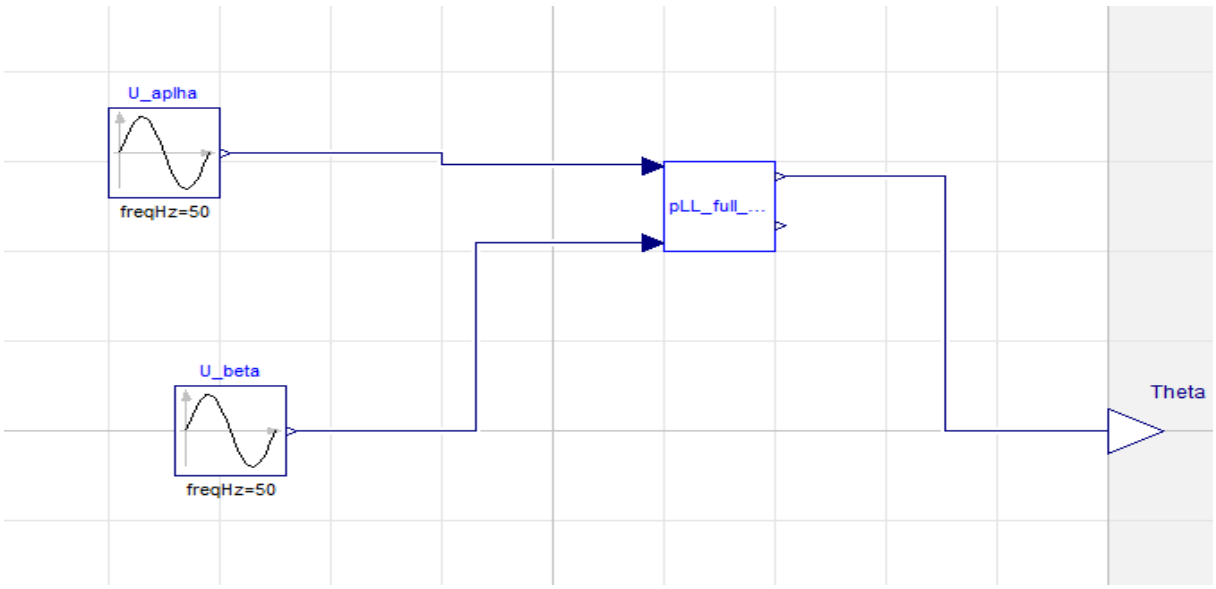

Figure 75 Modelica implementation of PLL test circuit

A time step of 1 µs and a simulation time of 0.1 second have been chosen in EMTP for the simulation of this circuit. Total number of Modelica intervals for the simulation is 100000.Figure 76 shows the output response of PLL block. There is a slight differnce in the peroid of the output obtained from EMTP and modelica signals.This has been caused by the FixedDelay block that was conidered as the equivalent of the Delay operation in EMTP.

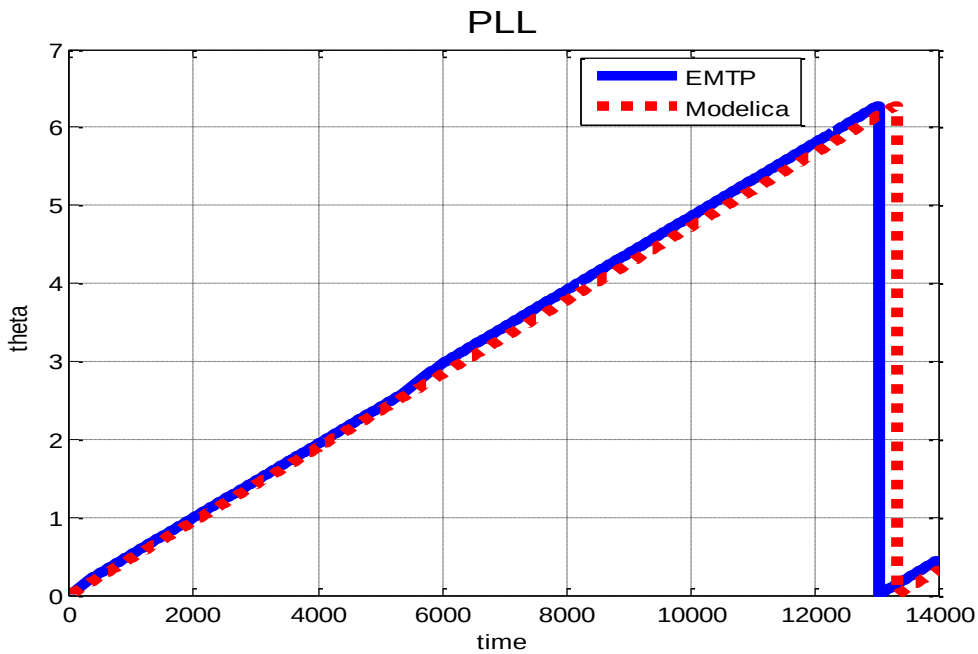

Figure 76 EMTP and Modelica response of PLL block

4.3. Validation of the control blocks

Control blocks like V/F control, Outer Control, Inner Control have PI or PID controllers inside. These blocks have been validated for transient inputs alongside the steady state constant inputs. Because in case of any disturbance there will be a sudden change in the voltage and current signal quantities which will cause a difference of the current and voltage signals with the respective reference signals. This difference would be the error signal that would enter into the PID or the PI controllers as an input and the controllers will respond accordingly as per the gain values of the controllers. For validation of the blocks step input has been used both in Modelica and EMTP as a source of transient input.

4.3.1. Outer Control

The input quantities for Outer control test are following:

Table 5: The input quantities for Outer control test

Vdc_meas	.11
U_primary_meas	.5
P_meas	.3
Q_meas	.37
Q_reference	0.3
Power_reference	0.3
Vdc_reference	.4
Kp_1	1
Ki_1	2
Kp_2	1
Ki_2	2
Kp_3	1
Ki_3	2
c14	1
c2	1

A time step of 10 ms and a simulation time of 10 second have been chosen in EMTP for the simulation with steady state input. Total number of Modelica intervals for the simulation is 1000. Again, a time step of 2 ms and a simulation time of 20 second have been chosen in EMTP for the simulation with transient input. Total number of Modelica intervals for the simulation is 10000. Figure 77 and Figure 78 show the EMTP and Modelica test circuit with the steady state input quantities.

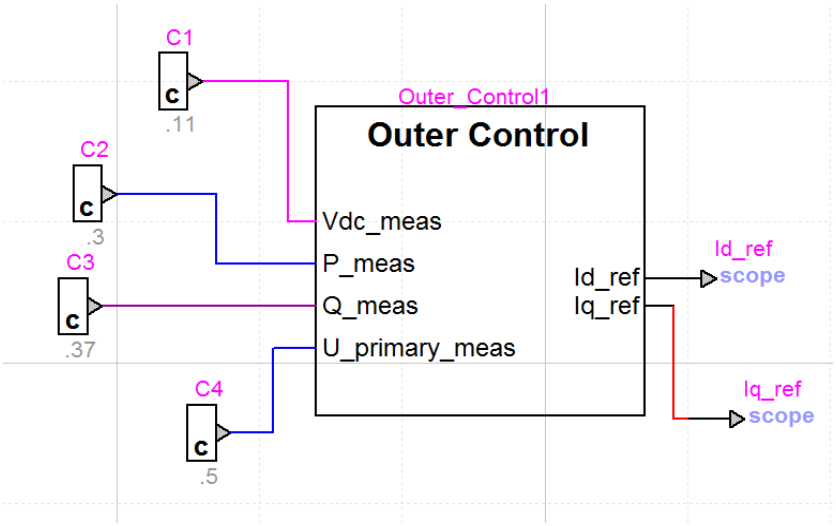

Figure 77 EMTP implementation of the Outer Control test circuit with steady state inputs

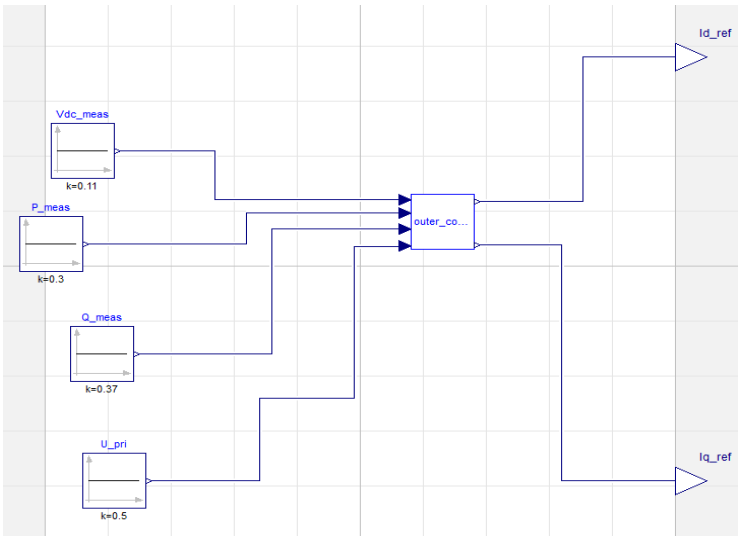

Figure 78 Modelica implementation of the Outer Control test circuit with steady state inputs

Figure 79 shows the output response of outer control block for steady state inputs.

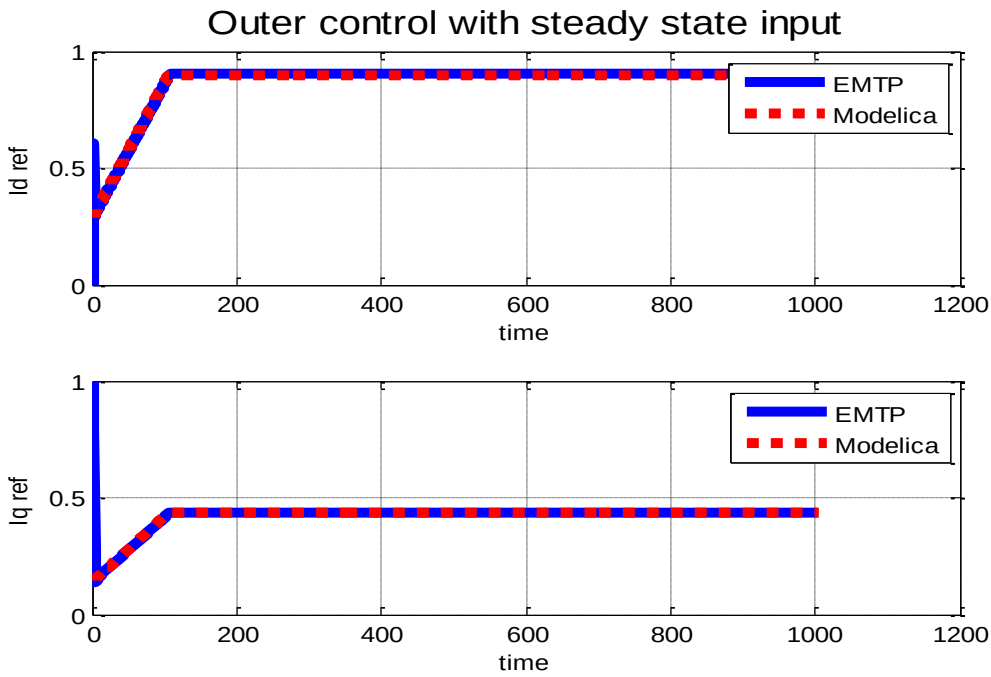

Figure 79 EMTP and Modelica response for Outer Control block for with steady state inputs

Figure 80 and Figure 81 shows the Modelica and EMTP test circuit for the transient inputs.

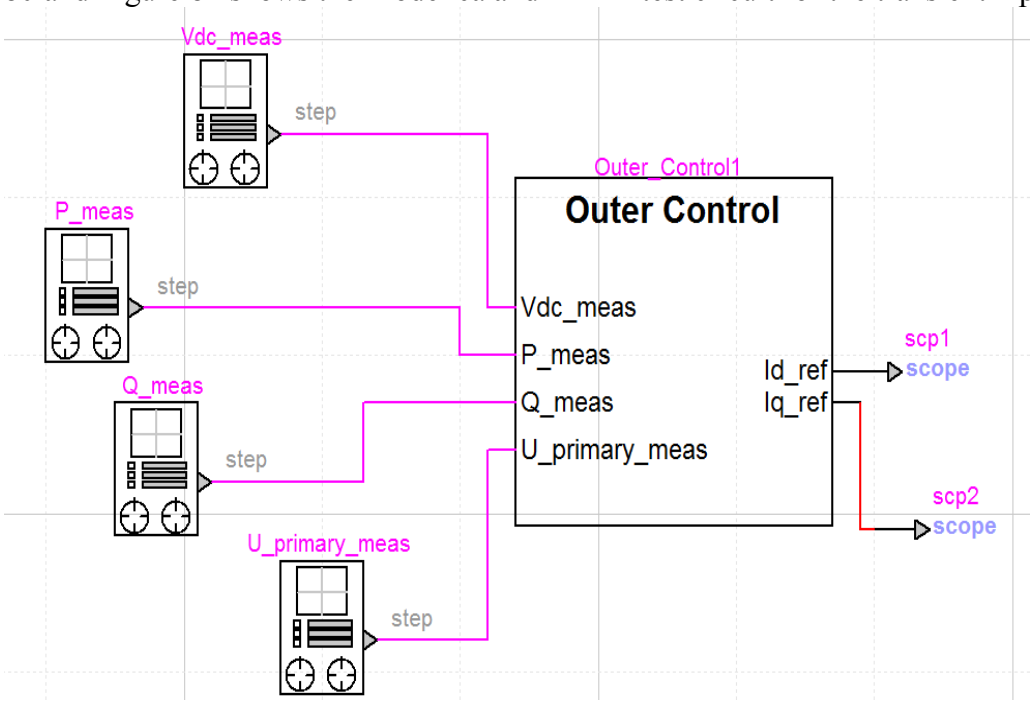

Figure 80 EMTP implementation of the Outer Control test circuit with transient inputs

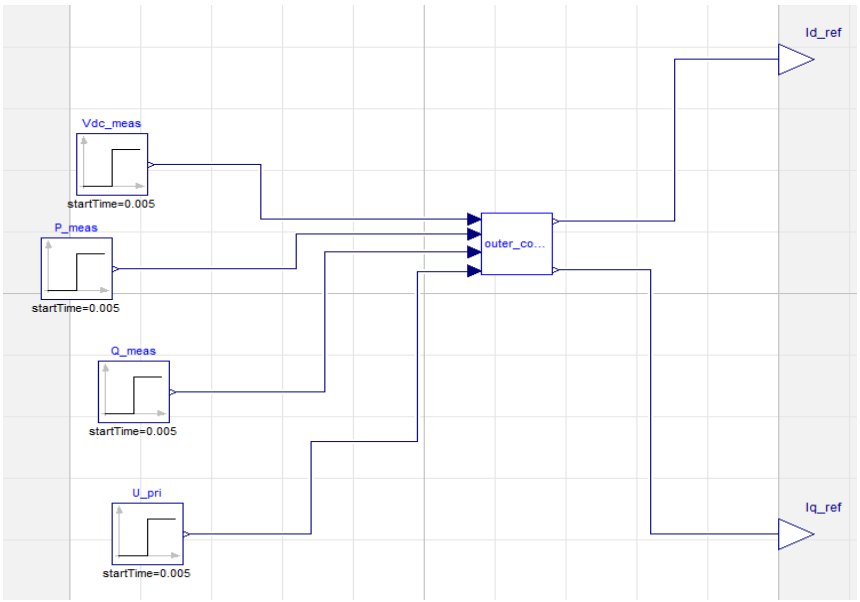

Figure 81 Modelica implementation of the Outer Control test circuit with transient inputs

The inputs of the transient test are presented in the next table:

Table 6 The input quantities for Outer control block for transient test

Quantity	Amplitude	Bias
Vdc_meas	.1	0.01
U_primary_meas	.4	0.1
P_meas	.2	0.1
Q_meas	.27	0.1
Q_reference	0.3	0.1
Power_reference	0.3	0.1
Vdc_reference	0.4	0.1

All the step signals used in the test are time shifted for 5 ms and they have the width of 1e15 ms. All the other controller quantities remain same as the test with the steady state input. Figure 82 shows the output response of outer control block to transient inputs.

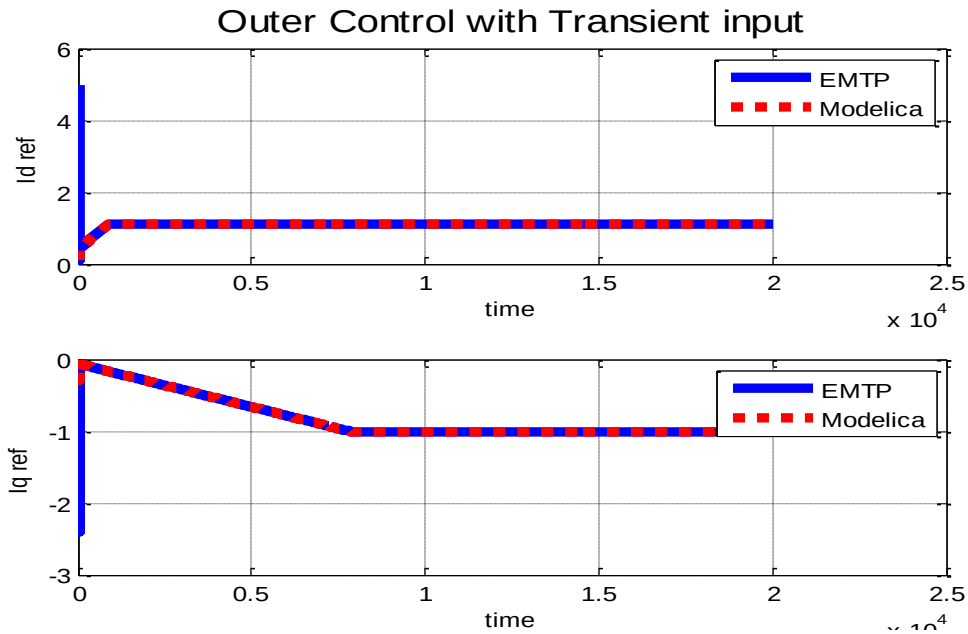

Figure 83 EMTP and Modelica response for Outer Control block for transient inputs

From the output result, it can be concluded that the designed Modelica block and the reference EMTP block of the Outer controller show almost similar performance for steady state and transient input.

4.3.2. Inner control test

The input quantities of Inner control for the steady state input are following:

Table 7 The input quantities of inner control for steady state input

id_ref	0.4
iq_ref	0.21
id	.5
iq	.4
Vd	0.2
Vq	0.3
Kp_1	0.5
Ki_1	0.7
Kp_2	0.5
Ki_2	0.7
X_trans	1
Xarm_equi	1

A time step of 1 ms and a simulation time of 20 second have been chosen in EMTP for the simulation with steady state input. Total number of Modelica intervals for the simulation is 20000. Again, a time step of 2 ms and a simulation time of 20 second have been chosen in EMTP for the simulation with transient input. Total number of Modelica intervals for the simulation is 10000.Figure 84 and Figure 85 show the EMTP and Modelica test circuit with the steady state input quantities.

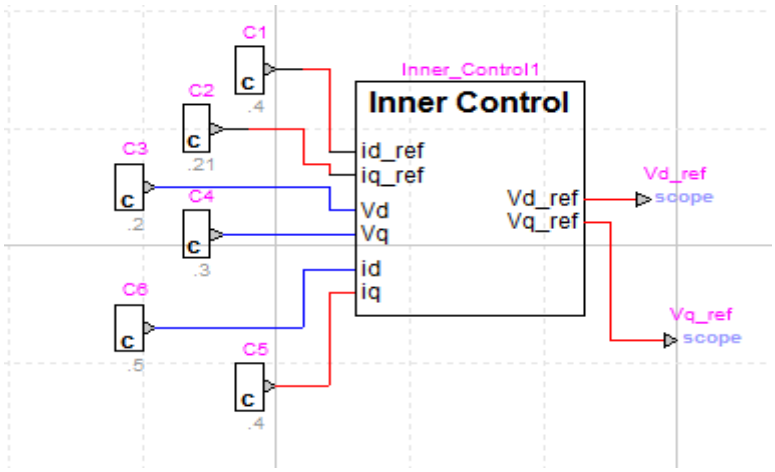

Figure 84 EMTP implementation of the Inner Control test circuit with steady state inputs

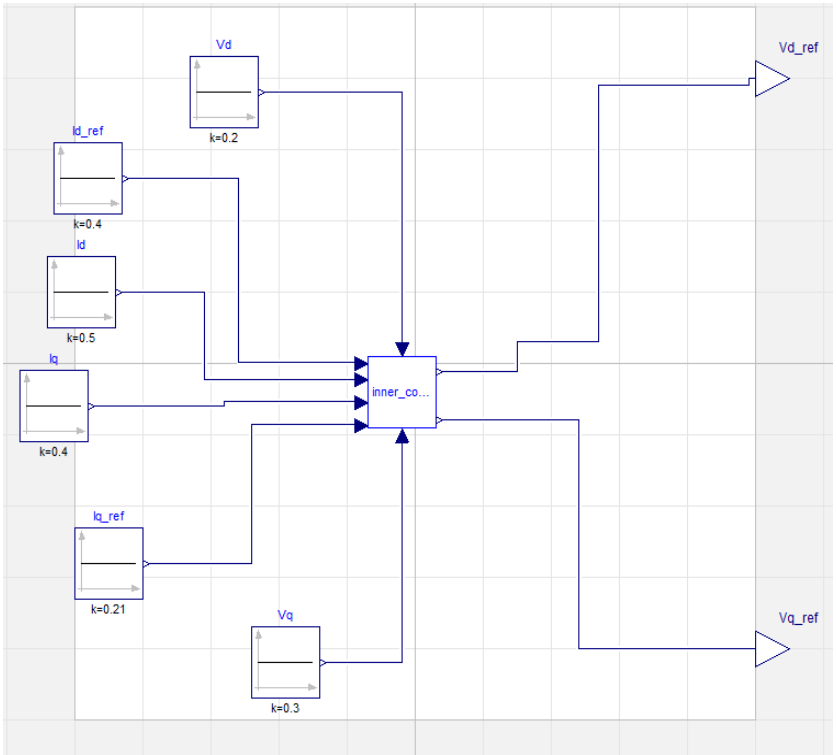

Figure 85 Modelica implementation of the Inner Control test circuit with steady state inputs

Figure 86 shows the output response of Inner control block for steady state inputs.

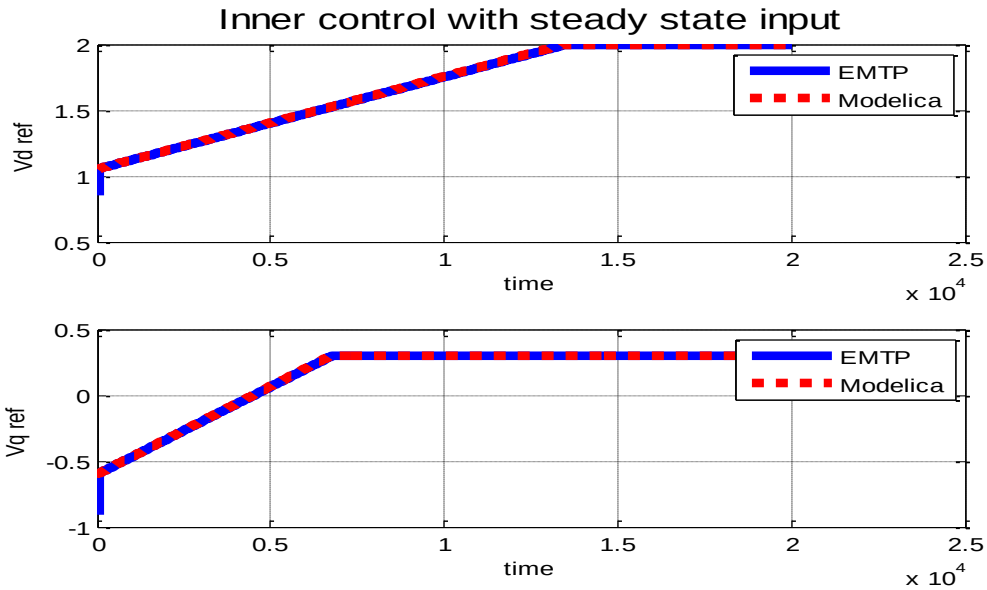

Figure 86 EMTP and Modelica response for Inner Control block for steady state inputs

The input quantities of Inner control of the transient inputs are following:

Table 8 The input quantities for inner control test

Quantities	Amplitude	Bias
id_ref	0.3	0.1
iq_ref	0.4	0.1
id	.5	0.1
Iq	.21	0.1
Vd	0.2	0.1
Vq	0.3	0.1

All the steps signals used in the test are time shifted for 5 ms and they have the width of 1e15 ms. All the other controller quantities remains same as the test with the steady state input test Figure 87 and Figure 88 show the EMTP and Modelica test circuit with the transient input quantities.

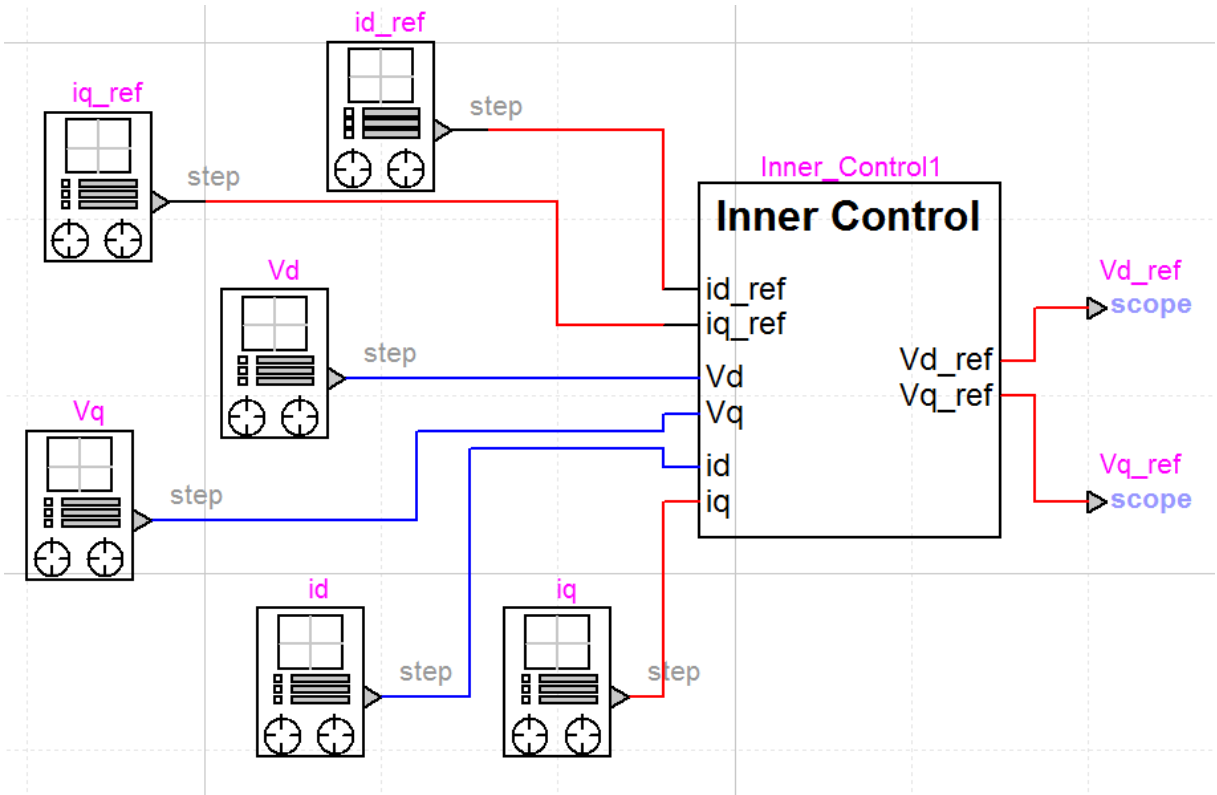

Figure 87 EMTP implementation of the Inner Control test circuit with transient inputs

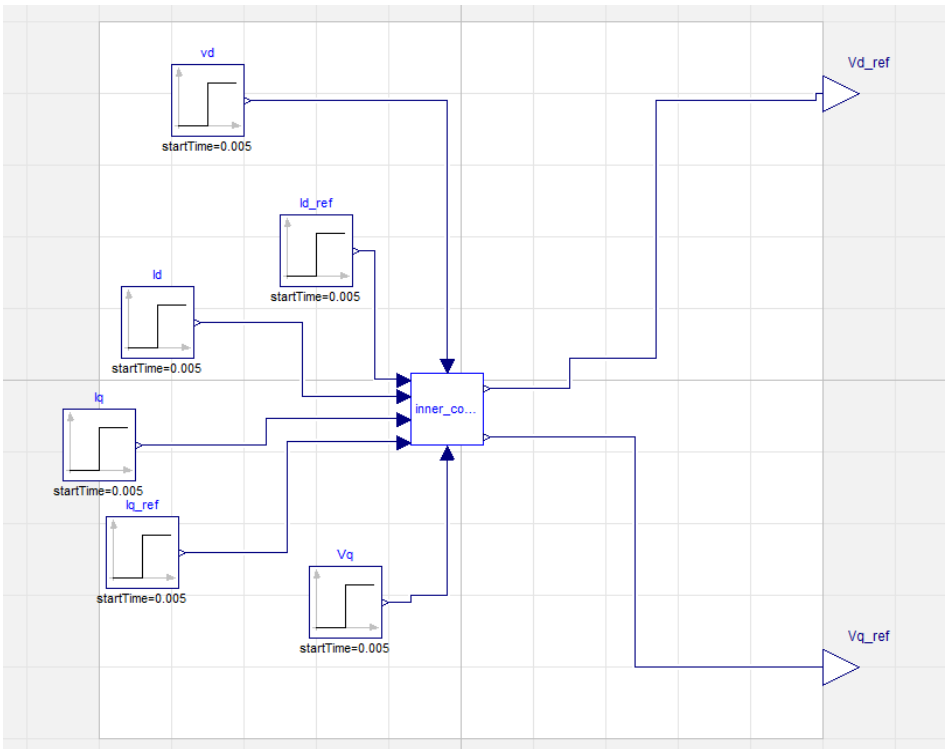

Figure 88 Modelica implementation of the Inner Control test circuit with transient inputs

Figure 89 shows the result of the Inner control block for transient inputs.

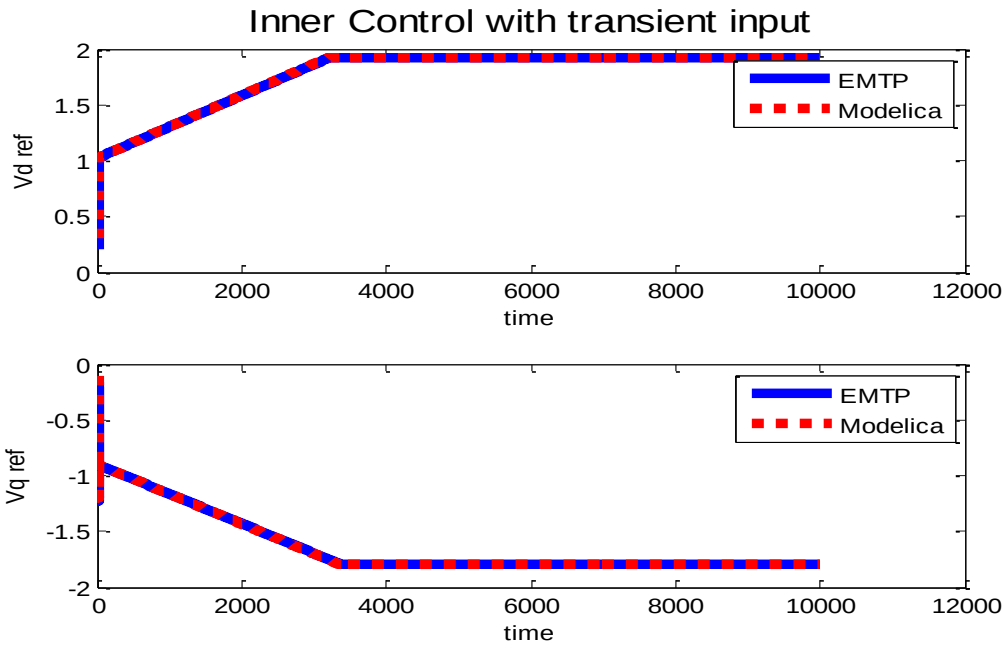

Figure 89 EMTP and Modelica response for Inner Control block for transient inputs

From the output result it can be concluded that the designed Modelica block and reference EMTP block for Inner Control performs the same operation.

4.3.3. V/F control

For V/F control test the steady state input quantities are U_meas=2.5 and theta=18. For the PID block the controller gains are Kp=1,Ki=1, Kd=2 and the initial output of the PID controller is 0.5

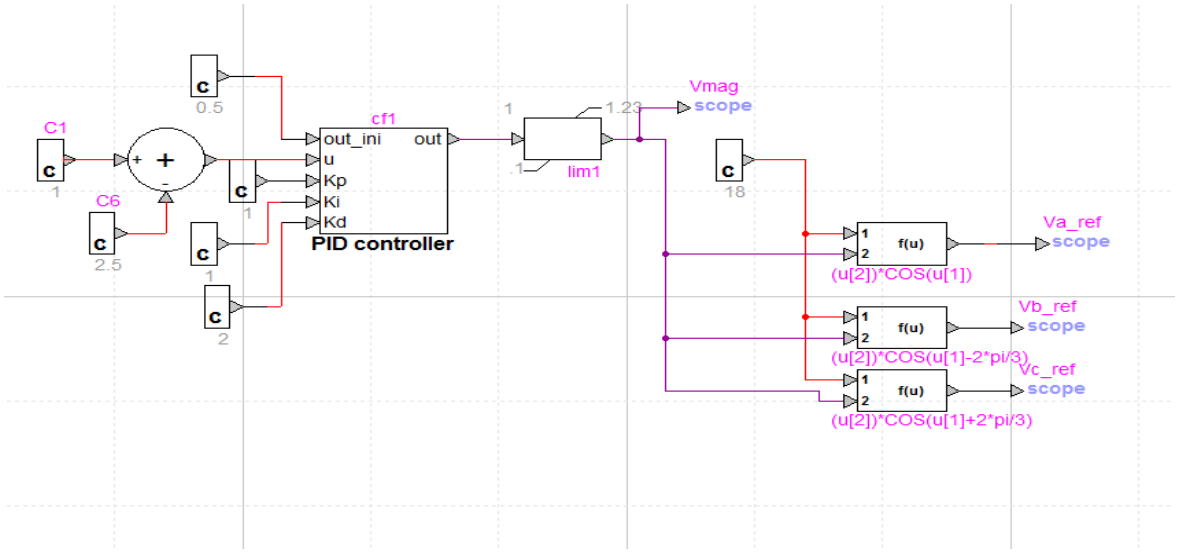

Figure 90 EMTP implementation of the V/F Control test circuit with steady state inputs

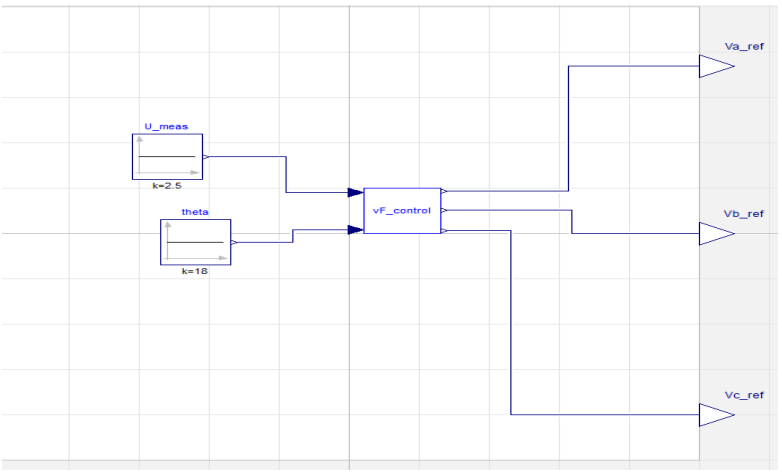

Figure 91 Modelica implementation of the V/F Control test circuit with steady state inputs

A time step of 2 ms and a simulation time of 10 second have been chosen in EMTP for the simulation with steady state input. Total number of Modelica intervals for the simulation is 5000. Again, a time step of 2 ms and a simulation time of 15 second have been chosen in EMTP for the simulation with transient input. Total number of Modelica intervals for the simulation is 7500.Figure 92 shows the output response of V/F control for constant inputs of U_meas and theta.

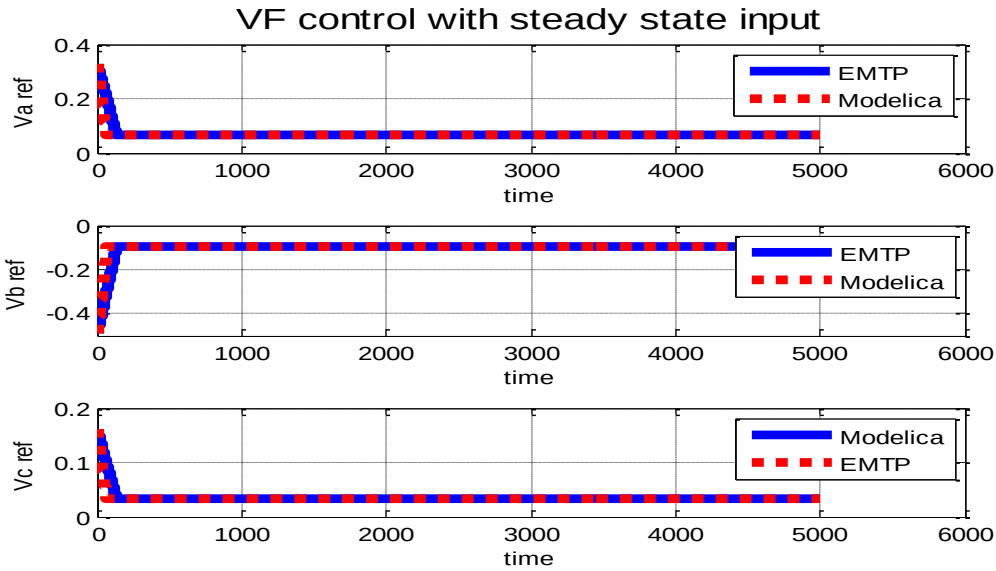

Figure 92 EMTP and Modelica response for V/F control for steady state inputs

For transient input the U_meas quantity changes to a step signal. The amplitude of the step signal is 0.3, width is 1e15 ms, bias is 0.1 and the signal is time shifted by 5 ms. The theta value is changed to 15. The control gains of PID controllers are: Kp=0. 5, Ki=0. 7, Kd=0. 3 and the initial output of the PID controller is 0.5.

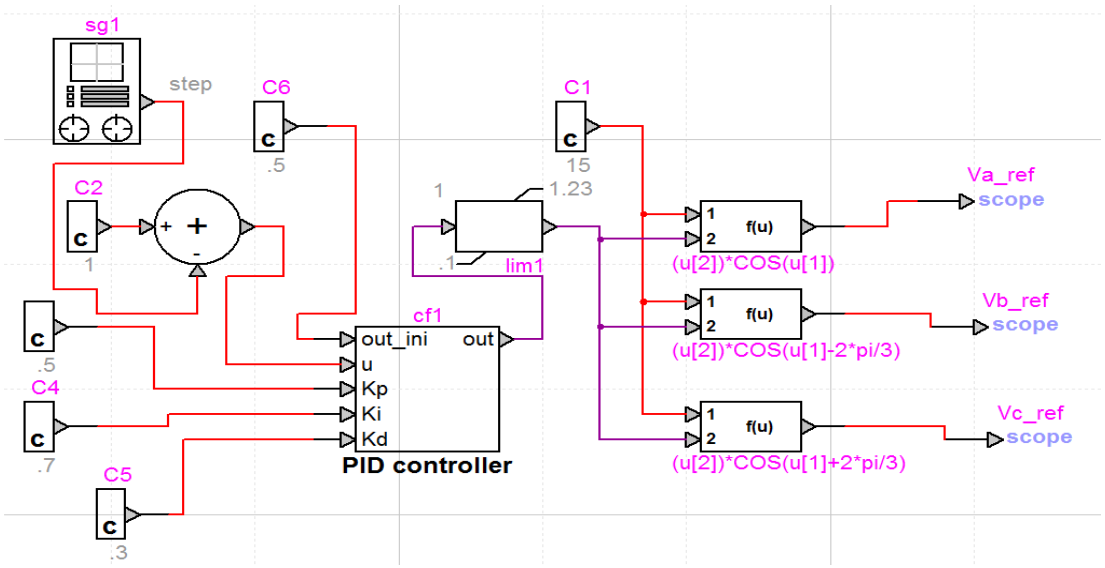

Figure 93 EMTP implementation of the V/F Control test circuit with transient inputs

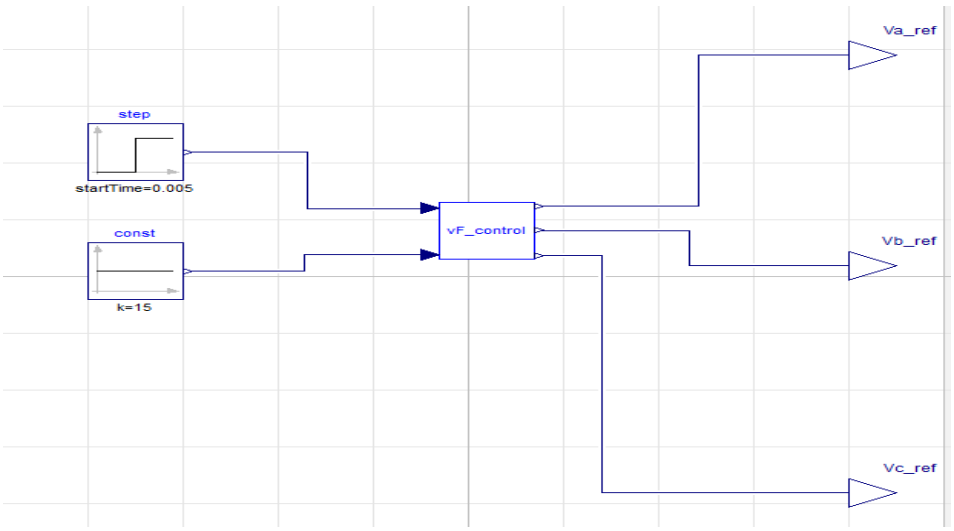

Figure 94 Modelica implementation of the V/F Control test circuit with transient inputs

Figure 95 shows the output response of V/F_control for transient inputs of U_meas.

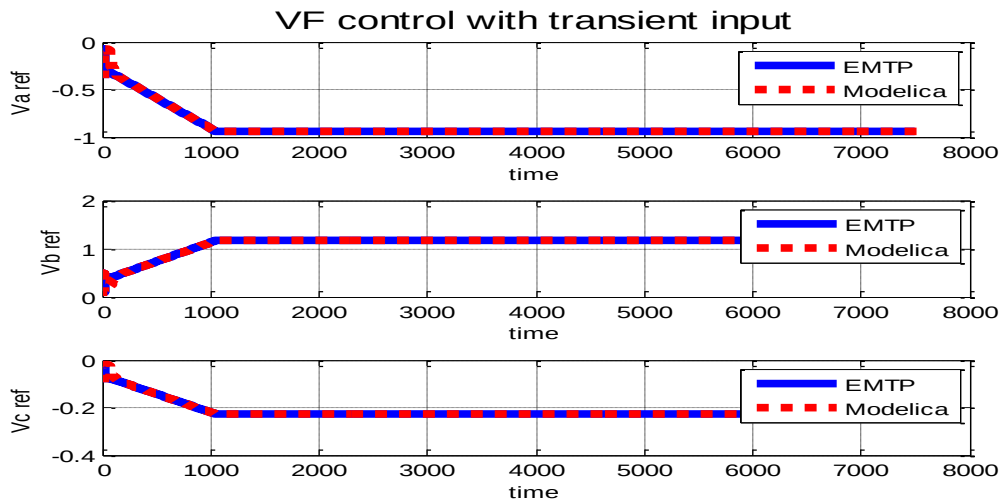

Figure 96 EMTP and Modelica response of V/F control for transient inputs

From the output results it can be concluded that the designed Modelica block and the reference EMTP block of the V/F controller show almost similar performance for steady state and transient input.

4.4. Validation of full Upper level control:

A simple test circuit is made for validating both the Upper-level control developed in Modelica and EMTP. Same values from the PID from Table 5 and Table 7 are used. A step source signal is used as the input of the Clarke transformation block. Per unit block and the filter were excluded in the test. For simplicity of the test setup all the Va and Ia, Vb and Ib, Vc and Ic quantities of the Clarke transformation block are kept similar. The theta value was taken from the Oscillator block.

Table 9 The input quantities for Upper level control test

Parameter	Step Amplitude	Bias
Ia	0.5	0.1
Ib	0.6	0.1
Ic	0.7	0.1
Vdc_meas	0.1	0.01

For all the step signals a time shift of 5 ms has been used.

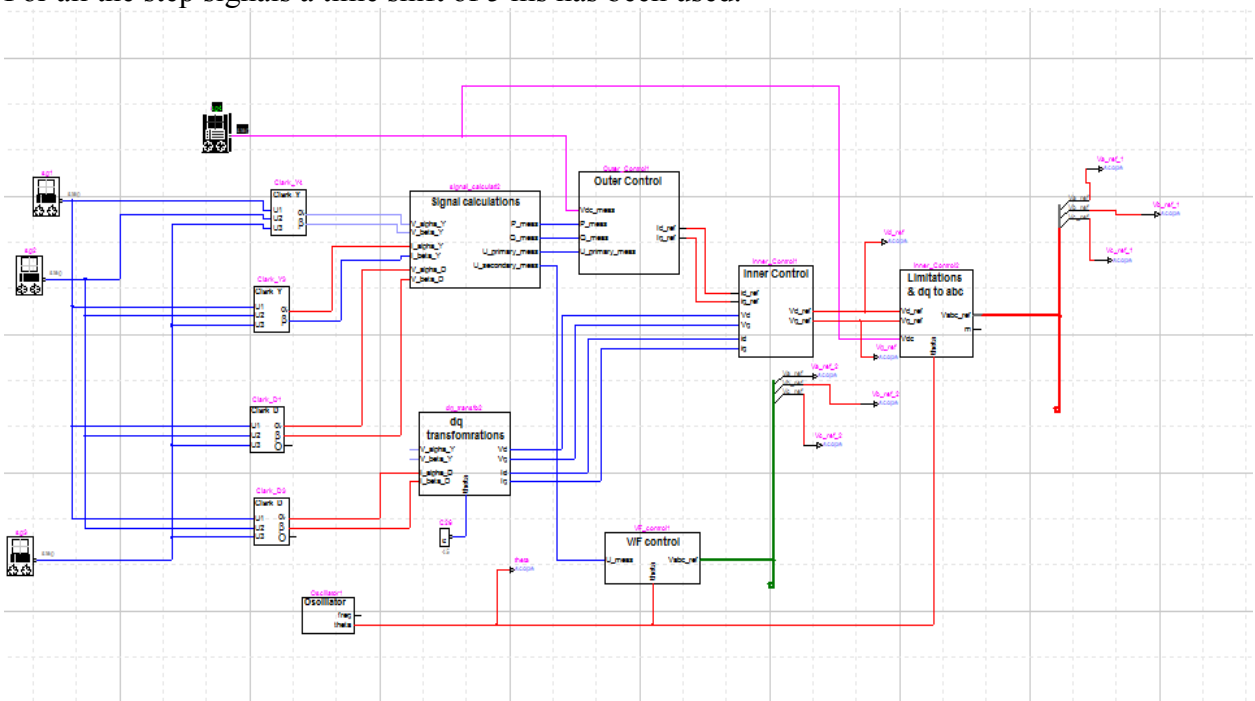

Figure 97 EMTP implementation of Upper level control test

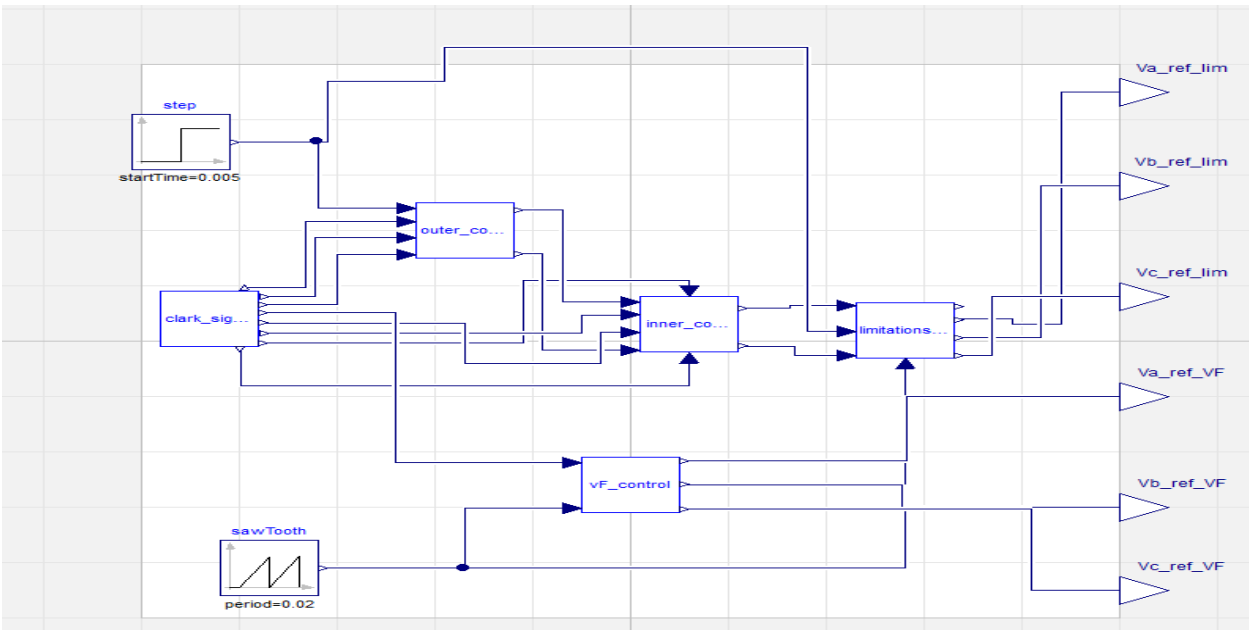

Figure 98 Modelica implementation of Upper level control test

A time step of 0.5 ms and a simulation time of 5 second have been chosen in EMTP for the simulation with steady state input. Total number of Modelica intervals for the simulation is 10000. Output reference voltage for both the V/F control and Limitaitons & dq to abc control block has been shown in Figure 99 and Figure 100

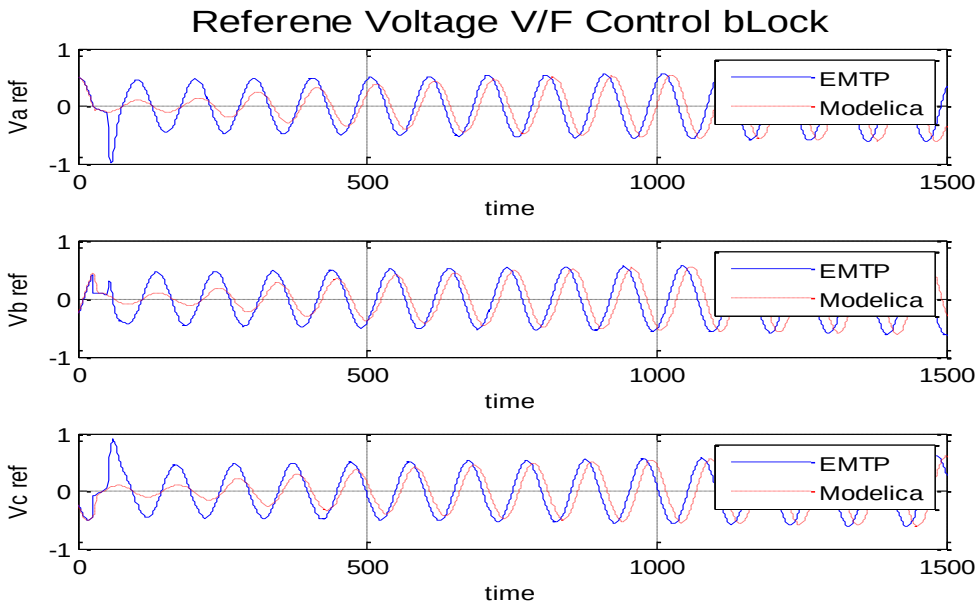

Figure 99: Three phase reference voltages output of of V/F control block

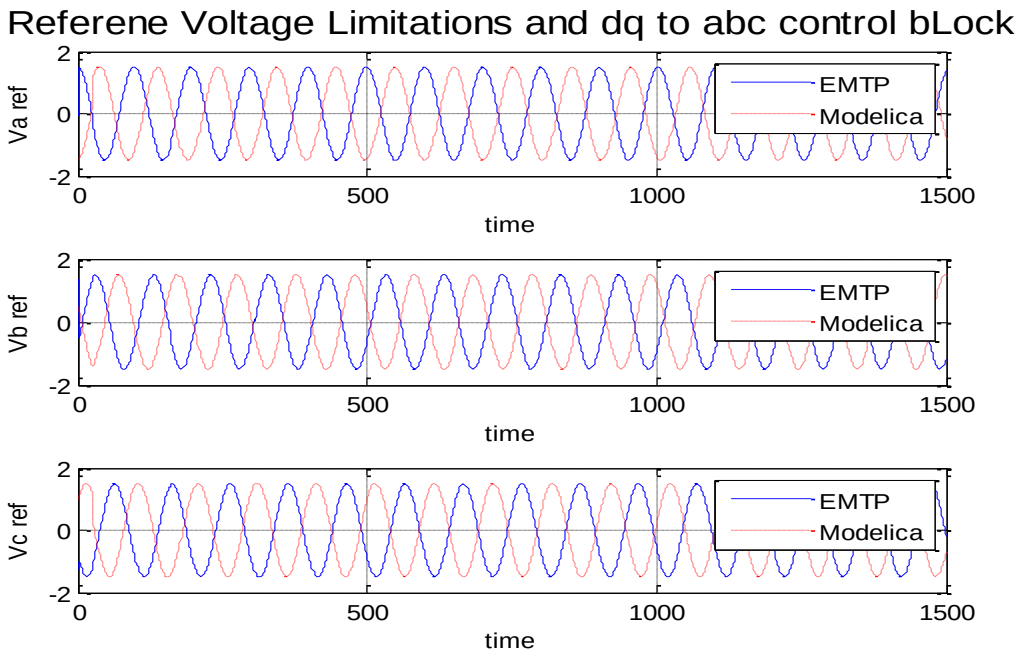

Figure 100: Three phase reference voltages output of Limitations and dq to abc control

The EMTP and Modelica output reference voltages have a time shift but the voltages are almost similar in magnitude. This time shift might be due to the small difference of phase angle theta

5. Conclusion and Future Work

5.1. Conclusion

The effort of this work has been focused on using Modelica language and the MSL to implement the AVM and Upper-level control for the reference model developed in EMTP. A lot of efforts have been made in understanding the process of simulation that EMTP follows. This process comprised the understanding of the implementation of the AVM and Upper-level control, the identification of their initial conditions and the identification of the different control strategies. The works have described the basic and important components that are used for the modeling of the AVM model. The implementation of some of the components has been straight forward since the Modelica Standard Library (MSL) already has these electrical components to work with, that can emulate the behavior as the electrical components available in EMTP. Other components have been developed in Modelica. With these models, the development of a library has been achieved.

Because of the lack of external measurements of these kinds of VSC-MMC models, the signal references for validation have been taken from simulations performed in EMTP. The entire AVM and Upper-level control models have been split into components and every component have been tested and validated. So, the work shows software to software validation from the simulations done with EMTP and Modelica to prove that Modelica and specifically, the simulation configuration from Dymola are valid tools to perform electromagnetic transient simulations.

5.2. Future work

The AVM model is only one representation of the VSC-MMC models for HVDC simulations. In this work, a library of components has been developed and new models will be implemented: A first task will be the implementation of other MMC topologies: (1) full detailed model, (2) detailed equivalent model and (3) switching function of arm model. And the implementation of the lower-level control model that is used in the previous MMC topologies.

6. References

[1].	Nikolas Flourentzou, Vassilios G. Agelidis and Georgios D. Demetriades, "VSC-Based HVDC Power Transmission Systems:An Overview," *IEEE TRANSACTIONS ON POWER ELECTRONICS*, VOL. 24, NO. 3, MARCH 2009.
[2].	A. Chieh, P. Panciatici, and J. Picard, "Power system modeling in Modelica for time-domain simulation," *2011 IEEE Trondheim PowerTech*, pp. 1–8, June 2011.
[3].	Modelica Design Group, Modelica - A Unified Object-Oriented Language for Physical Systems Modeling, Language Specification, 1999.
[4].	P. Fritzson, Introduction to Modeling and Simulation of Technical and Physical Systems with Modelica. John Wiley & Sons, 2011.
[5].	Jos Arrillaga "High Voltage Dirrect Current Transmission" 2nd Edition, *IEEE power and energy series* 29, 1998, ISBN 0 85296 941 4
[6].	L. Imsland, P. Kittilsen, and S. T.S., "Model-based optimizing control and estimation using modelica models," Modeling, Identification and Control, vol. 31, no. 3, pp. 107–121, 2010.
[7]	P. Fritzson, Principles of Object-Oriented Modeling and Simulation with Modelica 2.1, John Wiley & Sons, 2004

[8] K. J. Åström, H. Elmqvist, S. E. Mattsson, Evolution of continuous-time modeling and simulation, in: *Proceedings of the 12th European Simulation Multiconference, ESM'98, Society for Computer Simulation International,Manchester*, UK, 1998, pp. 9–18.

[9]. The Modelica Association, The Modelica Association Home Page, http://www.modelica.org (2007)

[10]. Roberto Rudervall, J.P. Charpentier, Raghuveer Sharma, "High Voltage Direct Current (HVDC)Transmission Systems Technology Review Paper" Presented at Energy Week 2000, Washington, D.C, USA, March 7-8, 2000

[11]. http://electrical-engineering-portal.com/8-main-disadvantages-of-hvdc-transmission. Accessed online on: 7th July, 2014.

[12]. BAHRMAN, M.P., 2008. HVDC transmission overview. *Transmission and Distribution Conference and Exposition*, 21-24 April 2008 Chicago. pp. 1-7.

[13]. D.M. Larruskain, I. Zamora, A.J. Mazon, 0.Abarrategui, J. Monasterio, "Transmission and
Distribution Networks: AC versus DC", *9th Spanish-Portuguese Congress on Electrical Engineering*, July 2005.

[14]. Andersen, B.R., "HVDC transmission-opportunities and challenges," *AC and DC Power Transmission, 2006. ACDC 2006. The 8th IEE International Conference on* , vol., no., pp.24,29, 28-31 March 2006 *The 8th IEE International Conference on, vol.*, no., pp. 24- 29, 28-31 March 2006.

[15]. Snigdha Sharma, "COMPARISON BETWEEN LCC AND VSC*", International Journal of Scientific Research Engineering & Technology (IJSRET)* Volume 2 Issue 7 pp 445-449 October 2013

[16]. Hani Saad and Jean Mahseredjian, “VSC MMC SATION MODELS MODULAR MULTILEVEL CONVERTER STATION in EMTP-RV Draft Report”.

[17] Sebastien Dennetiere, Hani Saad et al., “Guide for Development of Models for HVDC Converters in a HVDC Grid”, SEP 2013.

[18]. Peralta, J.; Saad, H.; Dennetiere, S.; Mahseredjian, J.; Nguefeu, S., "Detailed and Averaged Models for a 401-Level MMC–HVDC System," *Power Delivery, IEEE Transactions on* , vol.27, no.3, pp.1501,1508, July 2012

[19] Ama, N.R.N.; Martinz, F.O.; Matakas, L.; Kassab, F., "Phase-Locked Loop Based on Selective Harmonics Elimination for Utility Applications," *Power Electronics, IEEE Transactions on* , vol.28, no.1, pp.144,153, Jan. 2013

Appendix

Appendix.A: Validation of Park and Clarke inverse blocks

A.1

For the Park inverse test input values are: V_d=12, V_q= 32,V_O=42 and theta=23. A time step of 1 ms and a simulation time of 1 second have been chosen in EMTP for the simulation. Total number of Modelica intervals for the simulation is 1000.

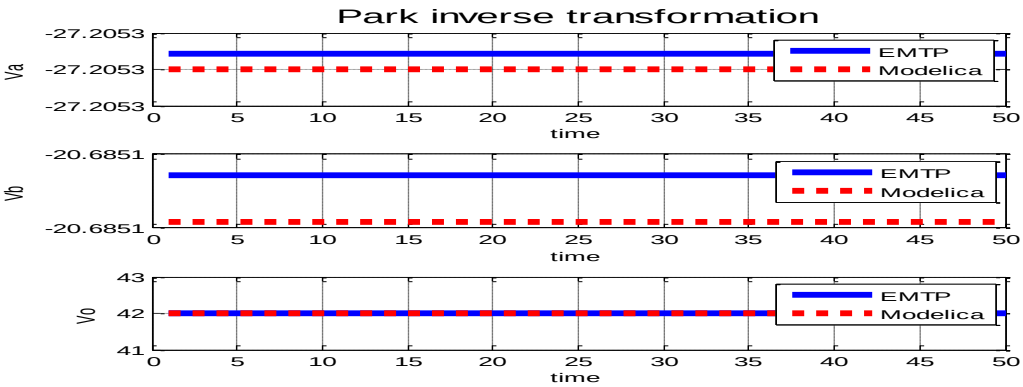

Figure 101 EMTP and Modelica response for Park inverse block

A.2

For the Clarke inverse test input values are : V_alpha=6, V_beta=8 and v_o=12. A time step of 1 ms and a simulation time of 1 second have been chosen in EMTP for the simulation. Total number of Modelica intervals for the simulation is 1000.

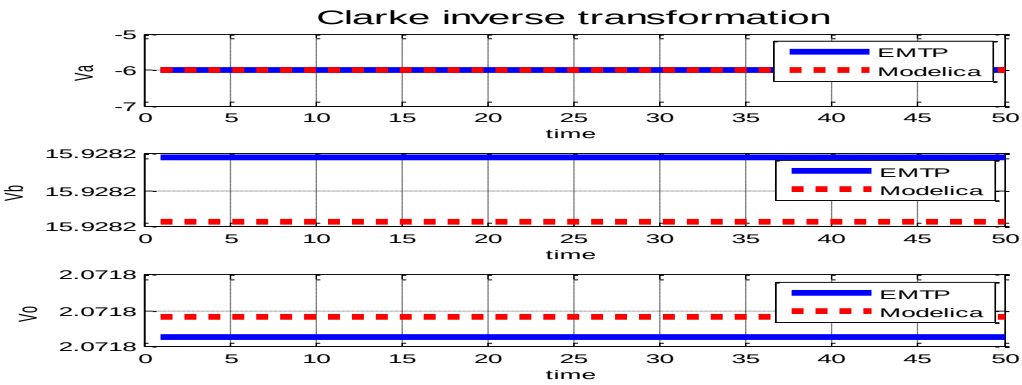

Figure 102 EMTP and Modelica response for Clarke inverse block

Appendix.B : Average Value Mean Frequency

The Avg.Value Mean Freq. block was tested for a constant frequency input of 50 Hz and a step input of Amplitude of 0.8, bias of 0.2 and time shift of 5 ms. A time step of 0.2 ms and a simulation time of 20 second have been chosen in EMTP for the simulation. Total number of Modelica intervals for the simulation is 100000.

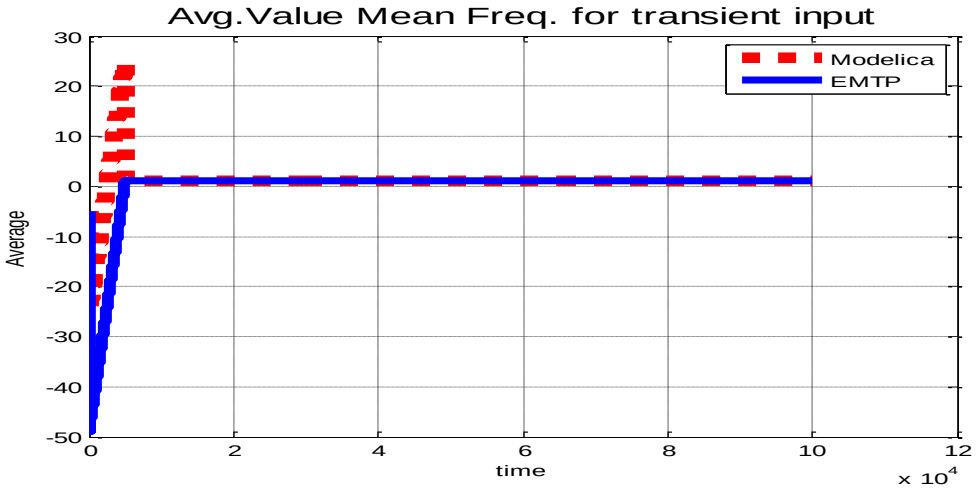

Figure 103 EMTP and Modelica response of the Avg.Value Mean.Freq block with transient inputs

Appendix.C: Implementation and validation of Selector block

C.1

Three reference voltages for Limitations & dq to abc are 10 and for V/F control are 5. For a 'False' value of the selector block the output of 'VF_control' block is selected as the output reference voltage. A time step of 0.2 ms and a simulation time of 1 second have been chosen in EMTP for the simulation. Total number of Modelica intervals for the simulation is 5000.

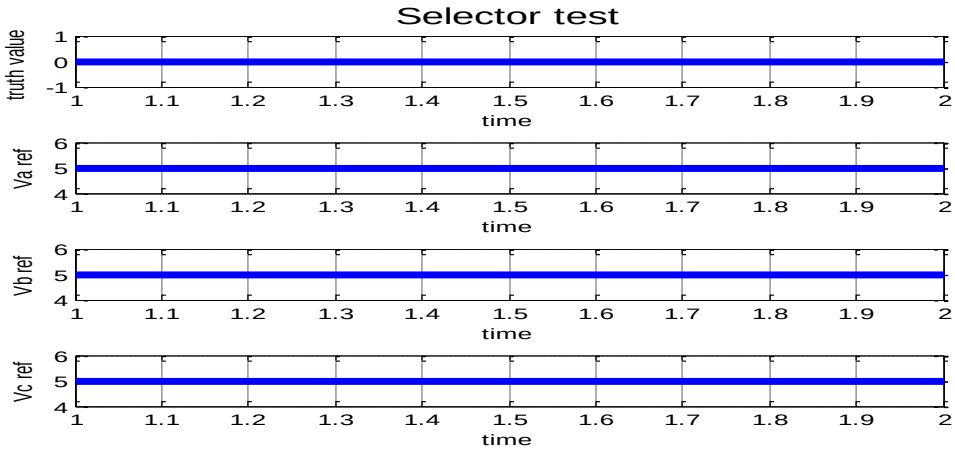

Figure 104 Modelica response of the Selector block with a 'False' truth value

C.2

For a 'True' value of the selector block the output of 'VF_control' block is selected as the output reference voltage.

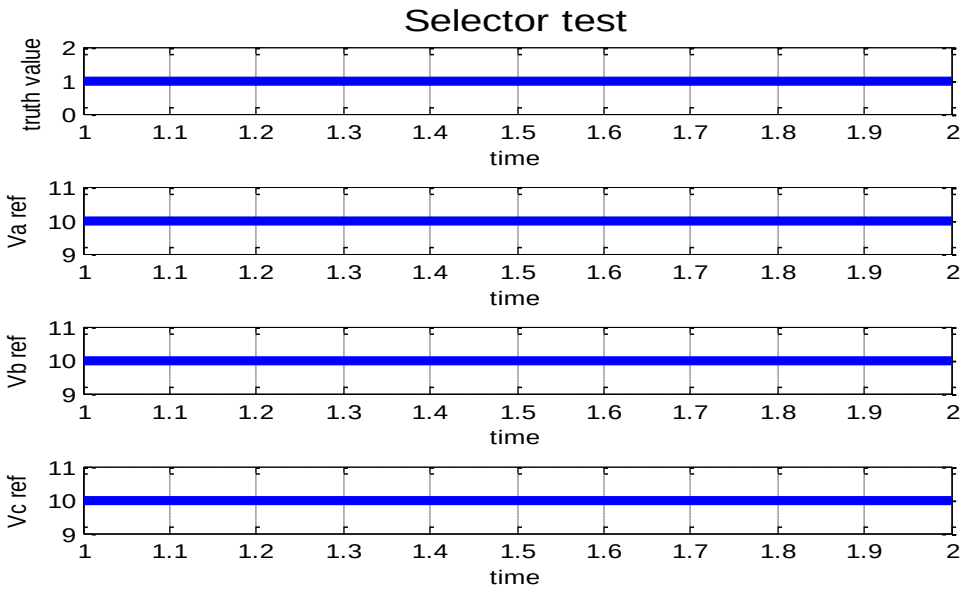

Figure 105 Modelica response of the Selector block with a 'True' truth value