ST. MARTIN'S ENGINEERING COLLEGE
An UGC Autonomous Institute
NBA & NAAC A+ Accredited
Dhulapally, Secunderabad-500 100
www.smec.ac.in

PROJECT REPORTS OF EEE

A

PROJECT REPORT

On

ADVANCED DRIVE SYSTEM FOR DC MOTOR USING MULTILEVEL DC/DC BUCK CONVERTER CIRCUIT

Submitted by

1) Mr.Godisela Mohan **Regd.No:17K81A0214**

2) Mr.M.VijayKumar **Regd.No:18K85A0218**

3) Mr.G.Ruthesh Yadav **Regd.No:18K85A0214**

4) Mr.M.Akshith Yadav **Regd.No:17K81A0243**

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY
IN

ELECTRICAL AND ELECTRONICS ENGINEERING

Under The Guidance of

Mr.G.Sridhar Babu,M.Tech,(Ph.D)

Associate Professor

DEPARTMENT OF ELECTRICAL AND ELECTRONICS NGINEERING

ST.MARTIN'S ENGINEERING COLLEGE

(An Autonomous Institute)

Dhulapally, Secunderabad – 500 100

JUNE 2021

BONAFIDE CERTIFICATE

This is to certify that the project entitled Advanced Drive System for DC Motor using Multilevel DC/DC Buck Converter Circuit, is being submitted by 1.Mr.Godisela Mohan Regd.No.17K81A0214 ,2.Mr.M.Vijay KumarRegd.No.18K85A0218,3.Mr.G.Ruthesh Yadav Regd.No.18K85A0214, 4.Mr.M.Akshith Yadav Regd.No.17K81A0243 in partial fulfillment of the requirement for the award of the degree of BACHELOR OF TECHNOLOGY IN ELECTRICAL AND ELECTRONICS ENGINEERING is recorded of bonafide work carried out by them. The result embodied in this report have been verified and found satisfactory.

Mr.G.Sridhar babu	Dr.N.Ramachandra
Associate Professor	Professor & HOD
Department of EEE	Department of EEE

Internal Examiner External Examiner

Place: Dhulapally

Date:

DECLARATION

We, the student of Bachelor of Technology in Department of ELECTRICAL AND ELECTRONICS ENGINEERING, session: 2017 – 2021, St. Martin's Engineering College, Dhulapally, Kompally, Secunderabad, hereby declare that work presented in this Project Work entitled ADVANCED DRIVE SYSTEM FOR DC MOTOR USING MULTILEVEL DC/DC BUCK CONVRTER CIRCUIT is the outcome of our own bonafide work and is correct to the best of our knowledge and this work has been undertaken taking care of Engineering Ethics. This result embodied in this project report has not been submitted in any university for award of any degree.

1. Mr.Godisela Mohan Regd.No.17K81A0214
2. Mr.M.Vijay Kumar Regd.No.18K85A0218
3. Mr.G.Ruthesh Yadav Regd.No.18K85A0214
4. Mr.M.Akshith Yadav Regd.No.17K81A0243

ACKNOWLEDGEMENTS

The satisfaction and euphoria that accompanies the successful completion of any task would be incomplete without the mention of the people who made it possible and whose encouragements and guidance have crowded effects with success.

First and foremost, we would like to express my deep sense of gratitude and indebtedness to **our College Management** for their kind support and permission to use the facilities available inthe Institute.

We especially would like to express my deep sense of gratitude and indebtedness to **Dr.P.Santosh Kumar Patra, our beloved Principal, St.Martin's Engineering College** for his invaluable encouragement, suggestions and support from an early stage of this research and providing me extraordinary experiences throughout the work. Above all, his priceless and meticulous supervision at each and every phase of work inspired me in innumerable ways.

We would like to express our gratitude to **Dr.N.Ramachandra**, **Head &Professor**, the Dept. of Electrical and Electronics Engineering for his advice, supervision, and the vital contribution as and when required during this research. We are proud to record that we had the opportunity to work with an exceptionally experienced Professor like him. The time spent with him will remain in my memory for years to come.

We would like to thank our supervisor, **Mr.G.Sridhar babu**, Associate professor, Dept. of EEE for his continuous support and valuable guidance for our research work.

We would like to express our gratitude to our **dept. project coordinators** Mr. K. V. GovardhanRao, Assistant Professor, Dept. of EEE and Mr. T. Penchalaiah, Assistant Professor, Dept of EEE for their involvement with originality has triggered and nourished my intellectual maturitythat will help me for a long time to come.

Finally, we are highly grateful and obliged to our college level Project Coordinators **Dr.Maalyadri, Professor in IT** and **Dr.R.Santhosh Kumar, Associate Professor in CSE** for their advice, supervision, and the vital contribution as and when required during this research work and for their support and co- operation that is difficult to express in words.

1.Mr.Godisela Mohan Regd.No.17K81A0214

2.Mr.M.Vijay Kumar Regd.No.18K85A0218

3.Mr.G.Ruthesh Yadav Regd.No.18K85A0214

3.Mr.G.Ruthesh Yadav Regd.No.18K85A0214

ABSTRACT

This project presents a new topology of clamped diode multilevel DC/DC buck power converter for a DC motor system. The proposed converter circuit consists of four cascaded MOSFET power switches with three clamping diodes and four voltage sources (voltage cells) connected in series. The main objective of the new topology is to reduce current ripples and torque ripples that are associated with hard switching of the traditional chopper circuit. When the voltage profile of this converter is applied on a DC motor, it positively affects the performance of the DC motor armature current and the generated dynamic torque. The output voltage of the proposed topology shows an adequate performance for tracking of reference voltage with small ripples that are normally reflected into smaller EMI noise. Moreover, it has been shown that the operation of the DC motor with the newly proposed chopper topology greatly decreases the motor armature current ripples and torque ripples by a factor equal to the number of the connected voltage cells.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF OUTPUT SCREENS

CHAPTER 1

INTRODUCTION

Nowadays, Direct Current (DC) motors are the main horse power of the most of the industrial process operations. These motors find a wide area of applications such as robotic motions, automatic manipulations, electric and hybrid vehicles, traction system, servo systems, rolling mills, and similar applications that require adequate process. The DC motors and their associate control and drive system are classified as the first choice compared to the available Alternating Current (AC) motors and their drive systems. The DC motor acquires this popularity due to many merits such as simplicity of its control and drive system compared to AC counterpart, linear variation of the torque and speed against applied armature voltage, wide controlled speed and wide controlled torque ranges, compact of size with high power efficiency for Permanent Magnet DC (PMDC) motors, and finally the overall low cost.

To control the DC motor rotor position, rotor speed, or the developed torque, the motor field current or the armature voltage is controlled to achieve the control goal. The armature terminal voltage through power electronic circuits is mostly used in the motor control system especially for the relatively high-power machines.

The application of pulse width modulation (PWM) with a large DC link voltage to the motor windings with hard switching strategy (as the case of traditional chopper circuit) causes an unsatisfactory dynamic behavior. The abrupt variations in the voltage and the associated change in the armature current corresponding to the PWM switching initiate a wide range of voltage and current harmonics, which lead to torque ripples and the associated mechanical vibrations and acoustic noise.

The mechanical vibration and noise in electric motors have become one of the most important factors for motor selection to do a certain task. The sound of the noise and the vibration in the motor are aroused mainly due to improper electromagnetic exciting forces that are continuously changed in time and space corresponding to the switching operation. This resultant variable-exciting force causes deformation in the mechanical structure and triggers the motor to vibrate

In a modern industrial situation, DC motor is widely used which is due to the low initial cost, excellent drive performance, low maintenance and the noise limit. As the electronic technology develops rapidly, its provide a wide scope of applications of high performance DC motor drives in areas such as rolling mills, electric vehicle tractions, electric trains, electric bicycles, guided vehicles, robotic manipulators, and home electrical appliances. DC motors have some control capabilities, which means that speed, torque and even direction of rotation can be changed at anytime to meet new condition. DC motors also can provide a high starting torque at low speed and it is possible to obtain speed control over a wide range.

So, the study of controlling DC motor is more practical significance. Control theory is an interdisciplinary branch of engineering and mathematics that deals with the behaviour of dynamical systems. For controlling a motor in any system, a controller is needed which is to give input to gate driver. For motor actuation, the microcontroller does not directly actuate the DC motor. It will have a device that known as gate driver which is function to drive the motor. For this system, it use motor driver as PWM amplifier to provide variable output voltage for controlling the speed of the motor and positive or negative voltage to control the direction of motor rotations. In real world, motor applications not only use the maximum speed of motor.

It maybe uses only 50% of its speed. So, the speed of the motor must be control. For some applications, motor is using not only one direction but with alternate direction to control a machine. In industrial field, some machine or robots cannot get in touch according to safety and the location of those things. The new method, which extensively used in motor controller, is pulse width modulation (PWM). PWM switching technique is a best method to control the speed of DC motor compare to another method. The duty cycle can be varied to get the variable output voltage. The concept of this system is same like DC-DC converter which is the output voltage depends on their duty cycle. Digital-to-analog conversion is not necessary because PWM itself is a signal that remains digital all the way from processor to control the overall system. By keeping the signal digital, noise effects are minimized unless there is a change from logic 1 to logic 0, which will make noise affect the digital signal. The Pulse-Width-Modulation (PWM) in microcontroller is used to control duty cycle of DC motor drive. PWM is an entirely different approach to controlling the speed of a DC motor. Power is supplied to the motor in square wave of constant voltage but varying pulse-width or duty cycle. Duty cycle refers to the percentage of one cycle during which duty cycle of a continuous train of pulses .

Since the frequency is held constant while the on-off time is varied, the duty cycle of PWM is determined by the pulse width. Thus the power increases duty cycle in PWM. A direct current (DC) motor converts DC electrical energy into mechanical energy. It produces a mechanical rotary action at the motor shaft where the shaft is physically coupled to a machine or other mechanical device to perform some type of work. DC motors are well suited for many industrial applications. For example, DC motors are used where accurate control of speed or position of the load is required and can be accelerate or decelerate quickly and smoothly. Plus, the direction easily reversed

CHAPTER 2

SPEED CONTROL OF DC MOTOR

DC MOTOR

There are two types of DC motors based on the construction such as self-excited, and separately excited. Similarly, self-excited motors classified into three types namely DC series motor, DC shunt motor, and DC compound motor. This article discusses an overview of the series motor, and the main function of this motor is to convert electrical energy to mechanical energy. The working principle of this motor mainly depends on electromagnetic law, which states that whenever a magnetic field is formed in the region of current carrying conductor & cooperates with an outside field, then the rotating motion can be generated. Once the series motor is started, then it will give utmost speed as well as torque slowly with high speed.

What is DC Series Motor?

The DC Series Motor is similar to any other motor because the main function of this motor is to convert electrical energy to mechanical energy. The operation of this motor mainly depends on the electromagnetic principle. Whenever the magnetic field is formed approximately, a current carrying conductor cooperates with an exterior magnetic field, and then a rotating motion can be generated.

Fig.2.1 DC Series Motor

Components used in DC Series Motor

The components of this motor mainly include the rotor (the armature), Commutator, stator, axle, field windings, and brushes. The fixed component of the motor is the stator, and it is built with two otherwise more electromagnet pole parts. The rotor includes the armature and the windings on the core allied to the Commutator. The power source can be connected toward the armature windings throughout a brush array allied to the Commutator.

The rotor includes a central axle for rotating, and the field winding must be able to hold high current due to the larger quantity of current throughout the winding, the larger will be the torque produced with the motor.

Therefore the motor winding can be fabricated with solid gauge wire. This wire does not permit a huge number of twists. The winding can be fabricated with solid copper bars because it assists in simple as well as efficient heat dissipation generated accordingly by a large amount of current flow during winding.

DC Series Motor Circuit Diagram

In this motor, field, as well as stator windings, are coupled in series by each other. Accordingly the armature and field current are equivalent.

Huge current supply straightly from the supply toward the field windings. The huge current can be carried by field windings because these windings have few turns as well as very thick. Generally, copper bars form stator windings. These thick copper bars dissipate heat generated by the heavy flow of current very effectively. Note that the stator field windings S1-S2 are in series with the rotating armature A1-A2.

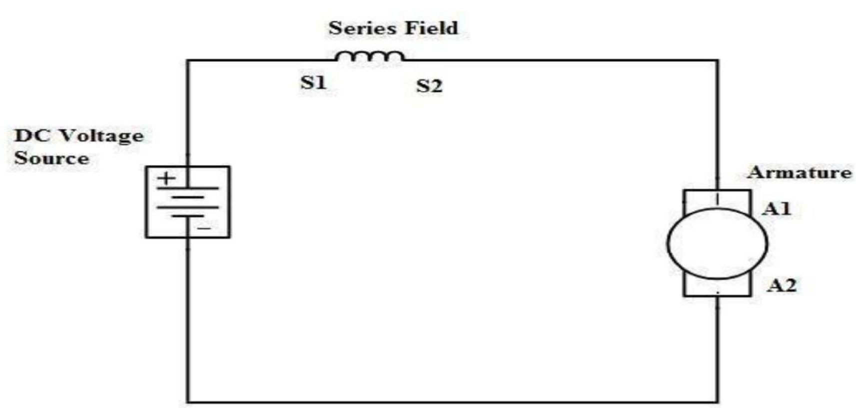


Fig.2.2 DC Series Motor Circuit Diagram

The strength of these magnetic fields provides the armature shafts with the greatest amount of torque possible. The large torque causes the armature to begin to spin with the maximum amount of power and the armature starts to rotate.

Electromagnetic motors

A coil of wire with a current running through it generates an electromagnetic field aligned with the center of the coil. The direction and magnitude of the magnetic field produced by the coil can be changed with the direction and magnitude of the current flowing through it.

A simple DC motor has a stationary set of magnets in the stator and an armature with one or more windings of insulated wire wrapped around a soft iron core that concentrates the magnetic field. The windings usually have multiple turns around the core, and in large motors there can be several parallel current paths. The ends of the wire winding are connected to a Commutator. The Commutator allows each armature coil to be energized in turn and connects the rotating coils with the external power supply through brushes. (Brushless DC motors have electronics that switch the DC current to each coil on and off and have no brushes.)

The total amount of current sent to the coil, the coil's size and what it's wrapped around dictate the strength of the electromagnetic field created.

The sequence of turning a particular coil on or off dictates what direction the effective electromagnetic fields are pointed. By turning on and off coils in sequence a rotating magnetic field can be created. These rotating magnetic fields interact with the magnetic fields of the magnets (permanent or electromagnets) in the stationary part of the motor (stator) to create a torque on the armature which causes it to rotate. In some DC motor designs the stator fields use electromagnets to create their magnetic fields which allow greater control over the motor.

At high power levels, DC motors are almost always cooled using forced air.

Different number of stator and armature fields as well as how they are connected provide different inherent speed/torque regulation characteristics. The speed of a DC motor can be controlled by changing the voltage applied to the armature. The introduction of variable resistance in the armature circuit or field circuit allowed speed control. Modern DC motors are often controlled by power electronics systems which adjust the voltage by "chopping" the DC current into on and off cycles which have an effective lower voltage.

Since the series-wound DC motor develops its highest torque at low speed, it is often used in traction applications such as electric locomotives, and trams. The DC motor was the mainstay of electric traction drives on both electric and diesel-electric locomotives, street-cars/trams and diesel electric drilling rigs for many years. The introduction of DC motors and an electrical grid system to run machinery starting in the 1870s started a new second Industrial Revolution. DC motors can operate directly from rechargeable batteries, providing the motive power for the first electric vehicles and today's hybrid cars and electric cars as well as driving a host of cordless tools. Today DC motors are still found in applications as small as toys and disk drives, or in large sizes to operate steel rolling mills and paper machines. Large DC motors with separately excited fields were generally used with winder drives for mine hoists, for high torque as well as smooth speed control using thyristor drives. These are now replaced with large AC motors with variable frequency drives.

If external mechanical power is applied to a DC motor it acts as a DC generator, a dynamo. This feature is used to slow down and recharge batteries on hybrid and electric cars or to return electricity back to the electric grid used on a street car or electric powered train line when they slow down. This process is called regenerative braking on hybrid and electric cars. In diesel electric locomotives they also use their DC motors as generators to slow down but dissipate the energy in resistor stacks. Newer designs are adding large battery packs to recapture some of this energy.

Speed control of DC series Motor

The speed of this type of motor is controlled by the following methods

- Armature resistance control
- Field control
- Tapped Field control

The most frequently used method is armature-resistance control method. Because in this method, the flux generated by this motor can be changed. The difference of flux can be attained by using the three methods like field diverters, armature diverter, and tapped field control.

Armature-resistance Control:

In the armature resistance control method, a changeable resistance can directly be connected in series through the supply. This can reduce the voltage which is accessible across the armature &

the speed drop. By altering the variable resistance value, any speed under the regular speed can be attained. This is the most general method used to control the DC series motor speed.

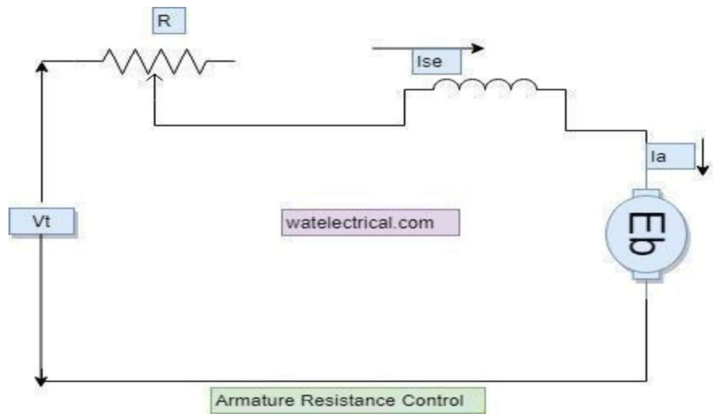


Fig.2.3

From the diagram, it is clear that as and when the resistance is adjusted the speed is varied. We know that in a series motor the field winding is in series and line and field current are the same. Due to this, the current flow depends upon the value of resistance. If more resistance is put in series with the armature, then the current flow will be less and vice-versa if less resistance is put in series.

The relationship between the speed and the back emf is given by **N E_b/ϕ**

Field Control Method

The circuit diagram that explains the field control method is shown in the figure below.

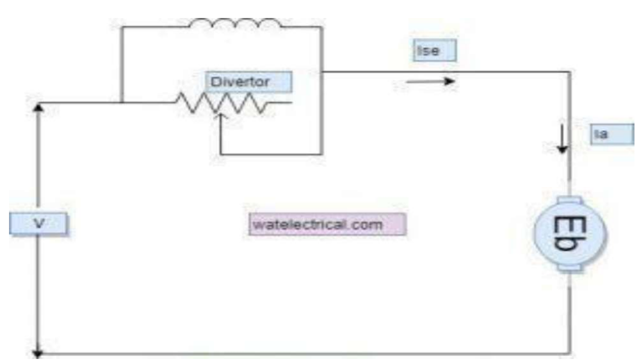


Fig no.2.4

Tapped diverter method

From the diagram, we can observe that a field diverter is connected across the field winding which is in series with the armature. The use of this diverter is to bypass the amount of flow armature current through the machine. As and when the armature current is varied we can able to vary the speed of the motor just like in armature resistance control. But the difference is, here we bypass the armature current by allowing some amount of current through the field winding as required. This is achieved by varying the resistance connected across field winding. If the diverter has maximum resistance, the current flows through the field winding and vice-versa if less resistance is connected.

Tapped Field Control Method

The circuit diagram that explains the Tapped field control method is shown in the figure below.

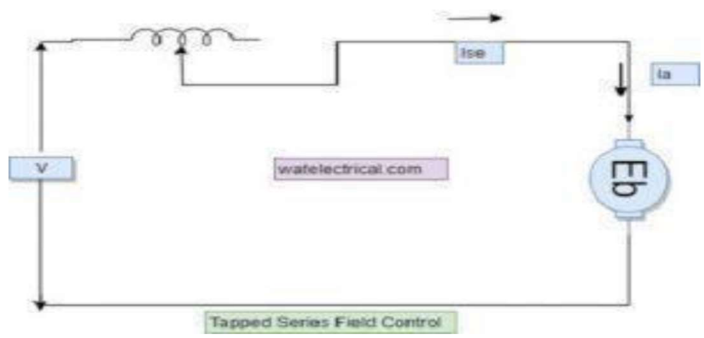


Fig.2.5

From the diagram, we can observe that a tapped series field resistance is connected in series with the armature. Just like the armature resistance control method, the series field is tapped at different points to vary the speed.

Speed Torque Characteristics of DC Series Motor

In general, for this motor, there are 3-characteristic curves are considered significant like Torque Vs. armature current, Speed Vs. armature current, & Speed Vs. torque. These three characteristics are determined by using the following two relations.

Ta aϕ.Ia
N aEb/ϕ

The above two equations can be calculated at the equations of emf as well as torque. For this motor, the back emf's magnitude can be given with the similar DC generator e.m.f equation like

Eb= Pϕ NZ / 60A. For a mechanism, A, P, and Z are stable, thus, N aEb/ϕ.

The **DC series motor torque equation** is,

Torque= Flux* Armature current

T = If * Ia

Here If= Ia, then the equation will become

T= Ia^2

Wound stators

The DC series motor torque (T) can be proportional to the Ia^2 (square of the armature current). In load test on dc series motor, the motor should be activated on load condition because if the motor can be activated on no load, then it will achieve an extremely high speed.

There are three types of electrical connections between the stator and rotor possible for DC electric motors: series, shunt/parallel and compound (various blends of series and shunt/parallel) and each has unique speed/torque characteristics appropriate for different loading torque profiles/signatures.

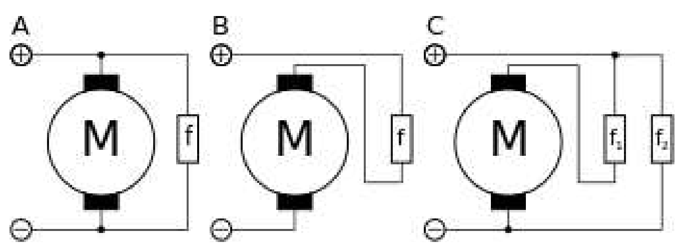


Fig .2.6

A field coil may be connected in shunt, in series, or in compound with the armature of a DC machine (motor or generator)

Series connection

A series DC motor connects the armature and field windings in series with a common D.C. power source. The motor speed varies as a non-linear function of load torque and armature current; current is common to both the stator and rotor yielding current squared (I^2) behavior. A series motor has very high starting torque and is commonly used for starting high inertia loads, such as trains, elevators or hoists. This speed/torque characteristic is useful in applications such as dragline excavators, where the digging tool moves rapidly when unloaded but slowly when carrying a heavy load.

A series motor should never be started at no load. With no mechanical load on the series motor, the current is low, the counter-Electro motive force produced by the field winding is weak, and so the armature must turn faster to produce sufficient counter-EMF to balance the supply voltage. The motor can be damaged by over speed. This is called a runaway condition.

Series motors called universal motors can be used on alternating current. Since the armature voltage and the field direction reverse at the same time, torque continues to be produced in the same direction. However they run at a lower speed with lower torque on AC supply when compared to DC due to reactance voltage drop in AC which is not present in DC. Since the speed is not related to the line frequency, universal motors can develop higher-than-synchronous speeds, making them lighter than induction motors of the same rated mechanical output. This is a valuable characteristic for hand-held power tools. Universal motors for commercial utility are usually of small capacity, not more than about 1 kW output. However, much larger universal motors were used for electric locomotives, fed by special low-frequency traction power networks to avoid problems with commutation under heavy and varying loads.

Shunt connection

A shunt DC motor connects the armature and field windings in parallel or shunt with a common D.C. power source. This type of motor has good speed regulation even as the load varies, but does not have the starting torque of a series DC motor. It is typically used for industrial, adjustable speed applications, such as machine tools, winding/unwinding machines and tensioners.

Compound connection

A compound DC motor connects the armature and fields windings in a shunt and a series combination to give it characteristics of both a shunt and a series DC motor. This motor is used when both a high starting torque and good speed regulation is needed. The motor can be connected in two arrangements: cumulatively or differentially. Cumulative compound motors connect the series field to aid the shunt field, which provides higher starting torque but less speed regulation. Differential compound DC motors have good speed regulation and are typically operated at constant speed.

DC Series Motor Advantages

The **advantages of the DC series motor** include the following.

- Vast starting torque
- Easy assembly and simple design
- Protection is easy
- Cost-effective

DC Series Motor Disadvantages

The disadvantages of DC series motor include the following.

- The motor speed regulation is fairly poor. When the load speed increases then the machine speed will decrease
- When the speed is increased, then the DC series motor's torque will be decreased sharply.
- This motor always needs the load before running the motor. So these motors are not suitable for where the motor's load is totally removed.

CHAPTER 3

PROPOSED CONVERTERS AND CONTROLLER CONFIGURATION

PROPOSED CONVERTER

The proposed block diagram of the multilevel chopper circuit (MLCC) for a DC motor drive system is shown in Fig. 1. This suggested system consists of the proposed MLCC block, H-bridge block in order to control the direction of the motor rotation, PMDC motor, in addition to many control blocks that arrange and synchronize the operation of the whole system. In this research work, the suggested multilevel chopper circuit (MLCC) is a 5-level power converter as illustrated in Fig. 2, it is composed of four controllable power switches

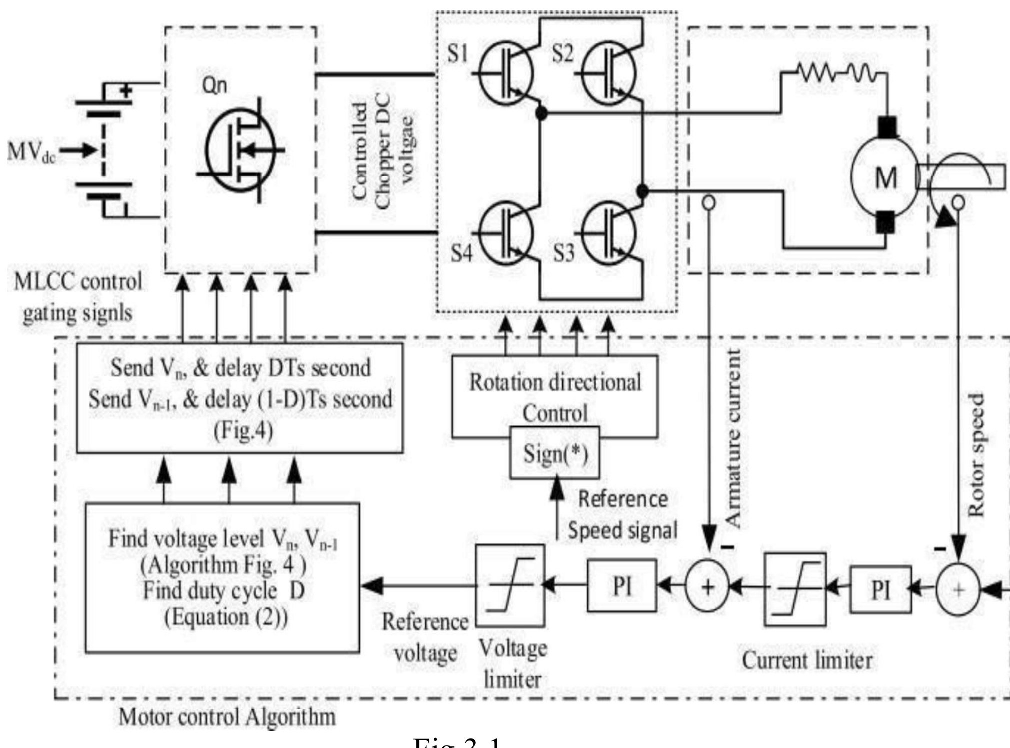


Fig.3.1

such as power MOSFET. The MLCC consists of three clamped diodes, (D1, D2 D3), preferably Schottky diodes and freewheeling diode DF: These diodes together with the power switches actualize the correct operation of the multilevel chopper circuit. The voltage of the sources VDC1, VDC2, VDC3 and VDC4 are of equal or different voltage values. These independent DC voltage sources could be cell storage batteries, solar cell units or any equivalent DC voltage sources.

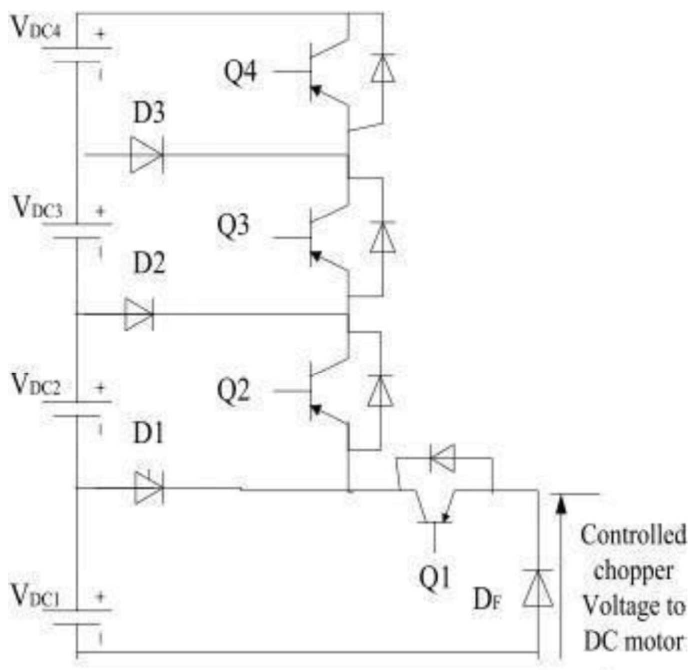


Fig.3.2

Controller design for any system needs knowledge about system behavior. Usually this involves a mathematical description of the relation among inputs to the process, state variables, and output. This description in the form of mathematical equations which describe behavior of the system (process) is called model of the system. This paper describes an efficient method to learn, analyze and simulation of power electronic converters, using system level nonlinear, and switched state- space models. The MATLAB/SIMULINK software package can be advantageously used to simulate power converters. This study aims at development of the models for all basic converters and studying its open loop response, so these models can be used in case of design of any close loop scheme. Also as a complete exercise a closed scheme case has been studied using cascaded control for a boost converter.

System modeling is probably the most important phase in any form of system control design work. The choice of a circuit model depends upon the objectives of the simulation. If the goal is to predict the behavior of a circuit before it is built. A good system model provides a designer with valuable information about the system dynamics. Due to the difficulty involved in solving general nonlinear equations, all the governing equations will be put together in block diagram form and then simulated using Matlab's Simulink program. Simulink will solve these nonlinear equations numerically, and provide a simulated response of the system dynamics.

To obtain a nonlinear model for power electronic circuits, one needs to apply Kirchhoff's circuit laws. To avoid the use of complex mathematics, the electrical and semiconductor devices must be represented as ideal components (zero ON voltages, zero OFF currents, zero switching times). Therefore, auxiliary binary variables can be used to determine the state of the switches. It must be ensure that the equations obtained by the use of Kirchhoff's laws should include all the permissible states due to power semiconductor devices being ON or OFF.

The steps to obtain a system-level modeling and simulation of power electronic converters are listed below.

1) Determine the state variables of the power circuit in order to write its switched state-space model, e.g., inductor current and capacitor voltage.

2) Assign integer variables to the power semiconductor (or to each switching cell) ON and OFF states.

3) Determine the conditions governing the states of the power semiconductors or the switching cell.

4) Assume the main operating modes of the converter (continuous or discontinuous conduction or both) or the modes needed to describe all the possible circuit operational modes. Then, apply Kirchhoff's laws and combine all the required stages into a switched state-space model, which is the desired system-level model.

5) Write this model in the integral form, or transform the differential form to include the semiconductors logical variables in the control vector: the converter will be represented by a set of nonlinear differential equations.

6) Implement the derived equations with "SIMULINK" blocks (open loop system simulation is then possible to check the obtained model).

7) Use the obtained switched space-state model to design linear or nonlinear controllers for the power converter.

8) Perform closed-loop simulations and evaluate converter performance.

9) The algorithm for solving the differential equations and the step size should be chosen before running any simulation. The two last steps are to obtain closed-loop simulations.

Each of the power electronic models represents subsystems within the simulation environment. These blocks have been developed so they can be interconnected in a consistent and simple manner for the construction of complex systems. The subsystems are masked, meaning that the user interface displays only the complete subsystem, and user prompts gather parameters for the entire subsystem. Relevant parameters can be set by double-clicking a mouse or pointer on each subsystem block, then entering the appropriate values in the resulting dialogue window.

To facilitate the subsequent simulation analysis and feedback controller verification, the pulse-width-modulation signal to control the ideal switch can also be built into the masked subsystem Fig. 9(a) and Fig. 9(b). For each converter to verity it's working in open loop configuration trigger pulses have been derived using a repeating sequence generator and duty cycle block. Function block compares the duty cycle and saw tooth from repeating sequence- derived trigger pulses are connected as an input to the switch control. Hence inputs for the masked subsystem are duty ratio and input voltage, and the outputs are chosen to be inductor current, capacitor voltage, and output voltage. When double-clicking the pointer on the masked subsystem, one enters parameter values of the switching converter circuit in a dialogue window. The intuitive signal flow interface in SIMULINK makes this mathematical model and its corresponding masked subsystem very easy to create.

PI Controller

A variation of Proportional Integral Derivative (PID) control is to use only the proportional and integral terms as PI control. The PI controller is the most popular variation, even more than full PID controllers. The value of the controller output u(t)u(t) is fed into the system as the manipulated variable input.

$$e(t) = SP - PV$$

$$u(t) = u_{bias} + K_c\, e(t) + \frac{K_c}{\tau_I} \int_0^t e(t)dt$$

The u_{bias} term is a constant that is typically set to the value of u(t)u(t) when the controller is first switched from manual to automatic mode. This gives "bump less" transfer if the error is zero when the controller is turned on. The two tuning values for a PI controller are the controller gain, KcKc and the integral time constant τI τI. The value of Kc Kc is a multiplier on the proportional error and integral term and a higher value makes the controller more aggressive at responding to errors away from the set point. The set point (*SP*) is the target value and process variable (*PV*) is the measured value that may deviate from the desired value. The error from the set point is the difference between the *SP* and *PV* and is defined as e(t)= SP−PVe(t) =SP-PV.

Digital controllers are implemented with discrete sampling periods and a discrete form of the PI equation is needed to approximate the integral of the error. This modification replaces the continuous form of the integral with a summation of the error and uses Δt as the time between sampling instances and ntnt as the number of sampling instances.

$$u(t) = u_{bias} + K_c\, e(t) + \frac{K_c}{\tau_I} \sum_{i=1}^{n_t} e_i(t)\Delta t$$

A P.I Controller is a feedback control loop that calculates an error signal by taking the difference between the output of a system, which in this case is the power being drawn from the battery, and the set point. The set point is the level at which we'd like to have our system running, ideally we'd like our system to be running near max power (990W) without causing the limiter to engage.

It is important to point out that due the the complexity of the electronic components within the circuit path(i.e ESC, power limiter, and motor) I was not able to accurately create model (transfer function) for the system. Having a transfer function would have allowed me to simulate the system in a software package such as MATLAB/Simulink and assist me in finding the right proportional and integral constant parameters for the controller. Unfortunately, due to the lack of a model, the parameters were obtained via a trial and error format. The figure above shows a software level block diagram of the P.I control algorithm. The controller receives a current and voltage measurement which it then uses to calculate the power being drained from the battery. Once the power is measured the error signal is calculated by taking the difference between the set point and the power measured. The error signal then goes into the P.I control loop where it gets multiplied by the proportional and integral constant. The output of the P.I control is a power value and in order to convert it to a quantity that is comparable to that of the control signal, it goes through a power to PWM signal converter. The adjusted PWM signal (output of PWM converter) then gets compared with the throttle signal, which is also a PWM signal, that is being sent by pilot,the least of the two gets sent to the controlled system. The controlled system block encompasses the battery, motor, speed controller, and limiter.

PI control is needed for non-integrating processes, meaning any process that eventually returns to the same output given the same set of inputs and disturbances. A P-only controller is best suited to integrating processes. Integral action is used to remove offset and can be thought of as an adjustable ubiasubias.

Common tuning correlations for PI control are the ITAE (Integral of Time-weighted Absolute Error) method and IMC (Internal Model Control). IMC is an extension of *lambda* tuning by accounting for time delay. The parameters KcKc, τpτp, and θpθp are obtained by fitting dynamic input and output data to a first-order plus dead-time (FOPDT) model.

An important feature of a controller with an integral term is to consider the case where the controller output u(t)u(t) saturates at an upper or lower bound for an extended period of time. This causes the integral term to accumulate to a large summation that causes the controller to stay at the saturation limit until the integral summation is reduced. Anti-reset windup is that the integral term does not accumulate if the controller output is saturated at an upper or lower limit.

Suppose that a driver of a vehicle set the desired speed set point to a value higher than the maximum speed. The automatic controller would saturate at full throttle and stay there until the driver lowered the set point. Suppose that the driver kept the speed set point higher than the maximum velocity of the vehicle for an hour. The discrepancy between the set point and the current speed would create a large integral term. If the driver then set the speed set point to zero, the controller would wait to lower the throttle until the negative error cancels out the positive error from the hour of driving. The automobile would not slow down but continue at full throttle for an extended period of time. This undesirable behavior is fixed by implementing anti-reset windup.

P-only Control

Simulate the behavior for using a P-only controller with Kc=2Kc=2 and Kc=0.5Kc=0.5. Implement a set point change from 0 to 10 and back in automatic mode (closed-loop). Include a plot of the error between the set point (*SP*) and process variable (*PV*). What happens with increased KcKc in terms of offset and oscillation?

PI Control

Configure the controller to add an integral term in addition to the proportional control with Kc=2Kc=2. Simulate the PI controller response with integral reset times τI=200,100,10τI=200,100,10. Include a plot of the integral of the error between the set point (*SP*) and process variable (*PV*) with anti-reset windup. Explain what happens and why.

Open Loop Response with Dead Time

Add dead time θp=100θp=100 as an input delay. Simulate the behavior for making a step change in manual mode from 0 to 10 (and back). Explain what happens in terms of oscillations.

P-only Control with Dead Time

With the dead time, simulate the response of a P-only controller with Kc=2Kc=2 and Kc=0.5Kc=0.5. Implement a set point change from 0 to 10 and back in automatic mode (closed-loop). Include a plot of the error between the set point (*SP*) and process variable (*PV*). What happens with increased KcKc in terms of offset and oscillation?

PI Control with Dead Time

Simulate the response of a PI controller with τI=200τI=200. Include a plot of the integral of the error between the set point (*SP*) and process variable (*PV*) with anti-reset windup. Explain what happens and why. Explain the results.

WHY PULSE WIDTH MODULATION

1. Cheap to make.
2. Little heat whilst working.
3. Low power consumption.
4. Can utilize very high frequencies (40-100 Khz is not uncommon.)
5. Very energy-efficient when used to convert voltages or to dim light bulbs.

6. High power handling capability
7. Efficiency up to 90%

a modulation technique used to encode a message into a pulsing signal. Although this modulation technique can be used to encode information for transmission, its main use is to allow the control of the power supplied to electrical devices, especially to inertial loads such as motors. In addition, PWM is one of the two principal algorithms used in photovoltaic solar battery chargers,[1] the other being MPPT.

The average value of voltage (and current) fed to the load is controlled by turning the switch between supply and load on and off at a fast rate. The longer the switch is on compared to the off periods, the higher the total power supplied to the load.

The PWM switching frequency has to be much higher than what would affect the load (the device that uses the power), which is to say that the resultant waveform perceived by the load must be as smooth as possible. Typically switching has to be done several times a minute in an electric stove, 120 Hz in a lamp dimmer, from few kilohertz (kHz) to tens of kHz for a motor drive and well into the tens or hundreds of kHz in audio amplifiers and computer power supplies.

The term duty cycle describes the proportion of 'on' time to the regular interval or 'period' of time; a low duty cycle corresponds to low power, because the power is off for most of the time. Duty cycle is expressed in percent, 100% being fully on.

The main advantage of PWM is that power loss in the switching devices is very low. When a switch is off there is practically no current, and when it is on and power is being transferred to the load, there is almost no voltage drop across the switch. Power loss, being the product of voltage and current, is thus in both cases close to zero. PWM also works well with digital controls, which, because of their on/off nature, can easily set the needed duty cycle.

PWM has also been used in certain communication systems where its duty cycle has been used to convey information over a communications channel.

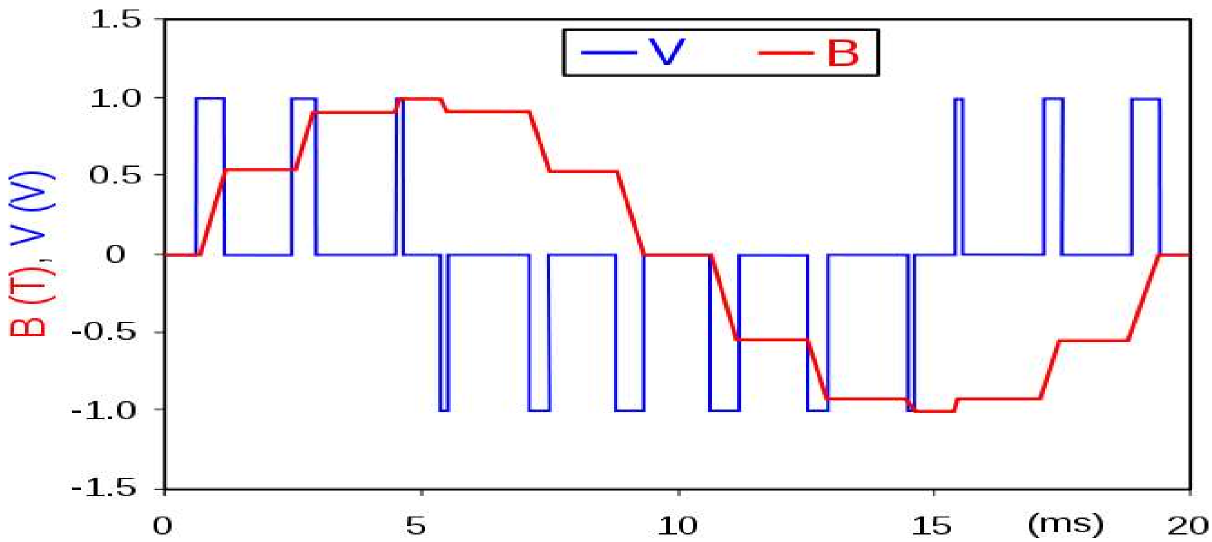


Fig 3.3 wave for combined positive and negative pulse

An example of PWM in an idealized inductor driven by a voltage source: the voltage source (blue) is modulated as a series of pulses that results in a sine-like current/flux (red) in the inductor. The blue rectangular pulses nonetheless result in a smoother and smoother red sine wave as the switching frequency increases. Note that the red waveform is the (definite) integral of the blue waveform.

Principle

Pulse-width modulation uses a rectangular pulse wave whose pulse width is modulated resulting in the variation of the average value of the waveform. If we consider a pulse waveform $f(t)$, with period T, low value y_{min} , a high value $y_{max}$and a duty cycle D (see figure 1), the average value of the waveform is given by:

$$\bar{y} = \frac{1}{T}\int_0^T f(t)\,dt.$$

As $f(t)$is a pulse wave, its value is y_{max} for $0 < t < D \cdot T$ and y_{min} for $D \cdot T < t < T$. The above expression then becomes:

$$\begin{aligned}\bar{y} &= \frac{1}{T}\left(\int_0^{DT} y_{max}\,dt + \int_{DT}^{T} y_{min}\,dt\right)\\ &= \frac{D \cdot T \cdot y_{max} + T\,(1-D)\,y_{min}}{T}\\ &= D \cdot y_{max} + (1-D)\,y_{min}.\end{aligned}$$

This latter expression can be fairly simplified in many cases where $y_{min} = 0$ as $\bar{y} = D \cdot y_{max}$. From this, it is obvious that the average value of the signal ($\bar{y}$) is directly dependent on the duty cycle D.

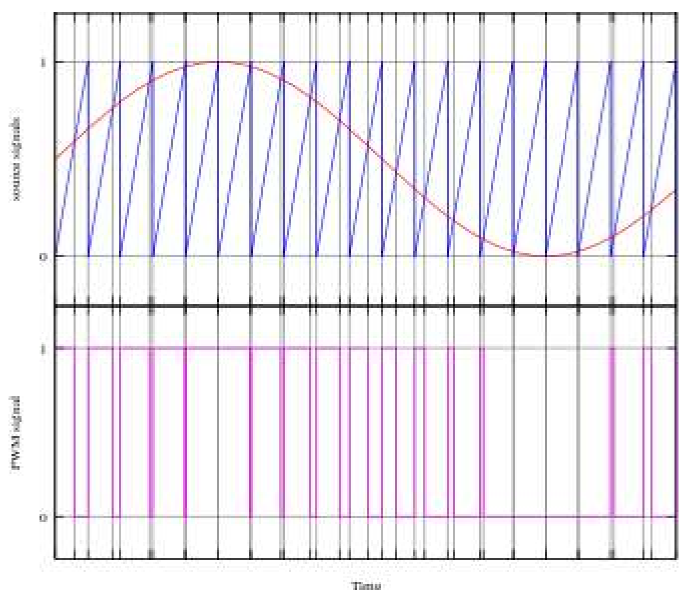


Fig. 3.4

Fig. 3.4 A simple method to generate the PWM pulse train corresponding to a given signal is the intersective PWM: the signal (here the red sine wave) is compared with a saw tooth waveform (blue). When the latter is less than the former, the PWM signal (magenta) is in high state (1). Otherwise it is in the low state (0).

The simplest way to generate a PWM signal is the intersective method, which requires only a sawtooth or a triangle waveform (easily generated using a simple oscillator) and a comparator. When the value of the reference signal (the red sine wave in figure 2) is more than the modulation waveform (blue), the PWM signal (magenta) is in the high state, otherwise it is in the low state.

The **PWM** is a technique which is used to drive the inertial loads since a very long time.The simple example of an inertial load is a motor. Apply the power to a motor for a very short period of time and then turn off the power: it can be observed that the motor is still running even after the power has been cut off from it. This is due to the inertia of the motor and the significance of this factor is that the continuous power is not required for that kind of devices to operate. A burst power can save the total power supplied to the load while achieving the same performance from the device as it runs on continuous power.

The **PWM technique** is use in devices like DC motors, Loudspeakers, Class -D Amplifiers, SMPS etc. They are also used in communication field as-well. The modulation techniques like AM, FM are widely used RF communication whereas the PWM is modulation technique is mostly used in Optical Fiber Communication (OFC).

As in the case of the inertial loads mentioned previously, the PWM in a communication link greatly saves the transmitter power. The immunity of the PWM transmission against the inter-symbol interference is another advantage. This article discusses the technique of generating a PWM wave corresponding to a modulating sine wave.

CHAPTER 4

PROPOSED CIRCUIT SIMULATION RESULTS

INTRODUCTION

Simulink is a software package for modeling, simulating, and analyzing dynamical systems. It supports linear and nonlinear systems, modeled in continuous time, sampled time, or a hybrid of the two. For modeling, Simulink provides a graphical user interface (GUI) for building models as block diagrams, using click-and-drag mouse operations. Models are hierarchical, so we can build models using both top-down and bottom-up approaches. We can view the system at a high level, then double-click on blocks to go down through the levels to see increasing levels of model detail. This approach provides insight into how a model is organized and how its parts interact. After we define a model, we can simulate it, using a choice of integration methods, either from the Simulink menus or by entering commands in MATLAB's command window. Using scopes and other display blocks, we can see the simulation results while the simulation is running. In addition, we can change parameters and immediately see what happens, for "what if" exploration.

The simulation results can be put in the MATLAB workspace for post processing and visualization. Simulink can be used to explore the behavior of a wide range of real-world dynamic systems, including electrical circuits, shock absorbers, braking systems, and many other electrical, mechanical, and thermodynamic systems.

Simulating a dynamic system is a two-step process with Simulink. First, we create a graphical model of the system to be simulated, using Simulink's model editor. The model depicts the time-dependent mathematical relationships among the system's inputs, states, and outputs. Then, we use Simulink to simulate the behavior of the system over a specified time span. Simulink uses information that you entered into the model to perform the simulation.

BLOCK DIAGRAM

A Simulink block diagram is a pictorial model of a dynamic system. It consists of a set of symbols, called blocks, interconnected by lines. Each block represents an elementary dynamic system that produces an output either continuously (a continuous block) or at specific points in time (a discrete block). The lines represent connections of block inputs to block outputs. Every block in a block diagram is an instance of a specific type of block. The type of the block determines the relationship between a block's outputs and its inputs, states, and time. A block diagram can contain any number of instances of any type of block needed to model a system. Blocks represent elementary dynamic systems that Simulink knows how to simulate. A block comprises one or more of the following:

1) A set of inputs,
2) A set of states, and
3) A set of outputs.

A block's output is a function of time and the block's inputs and states (if any). The specific function that relates a block's output to its inputs, states, and time depends on the type of block of which the block is an instance. Continuous Versus discrete Blocks Simulink's standard block set includes continuous blocks and discrete blocks. Continuous blocks respond continuously to continuously changing input. Discrete blocks, by contrast, respond to changes in input only at integral multiples of a fixed interval called the block's sample time. Discrete blocks hold their output constant between successive sample time hits. Each discrete block includes a sample time parameter that allows you to specify its sample rate. The Simulink blocks can be either continuous or discrete, depending on whether they are driven by continuous or discrete blocks. A block that can be either discrete or continuous is said to have an implicit sample rate. The implicit sample time is continuous if any of the block's inputs are continuous. The implicit sample time is equal to the shortest input sample time if all the input sample times are integral multiples of the shortest time. Otherwise, the input sample time is equal to the fundamental sample time of the inputs, where the fundamental sample time of a set of sample times is defined as the greatest integer divisor of the set of sample times.

Simulink can optionally color code a block diagram to indicate the sample times of the blocks it contains, e.g., black (continuous), magenta (constant), yellow (hybrid), red (fastest discrete), and so on. The block contains block name, icon, and block library that contain the block, the purpose of the block

SIMULINK BLOCK LIBRARIES

Simulink organizes its blocks into block libraries according to their behavior.

1) The Sources library contains blocks that generate signals.
2) The Sinks library contains blocks that display or write block output.
3) The Discrete library contains blocks that describe discrete-time components.
4) The Continuous library contains blocks that describe linear functions.
5) The Math library contains blocks that describe general mathematics functions.
6) The Functions & Tables library contains blocks that describe general functions and table look-up operations.
7) The Nonlinear library contains blocks that describe nonlinear functions.
8) The Signal & Systems library contains blocks that allow multiplexing and demultiplexing, implement external input/output, pass data to other parts of the model, and perform other functions.
9) The Subsystems library contains blocks for creating various types of subsystems.
10) The Block sets and Toolboxes library contains the Extras block library of specialized blocks.

SUB SYSTEMS

Simulink allows to model a complex system as a set of interconnected subsystems each of which is represented by a block diagram. We create a subsystem using Simulink's Subsystem block and the Simulink model editor. We can embed subsystems with subsystems to any depth to create hierarchical models. We can create conditionally executed subsystems that are executed only when a transition occurs on a triggering or enabling input.

SOLVERS

Simulink simulates a dynamic system by computing its states at successive time step solver a specified time span, using information provided by the model. The process of computing the successive states of a system from its model is known as solving the model. No single method of solving a model suffices for all systems. Accordingly, Simulink provides a set of programs, known as solvers, that each embody a particular approach to solving a model. The Simulation Parameters dialog box allows us to choose the solver most suitable for our model.

Fixed-Step and Variable-Step Solvers

Fixed-step solvers solve the model at regular time intervals from the beginning to the end of the simulation. The size of the interval is known as the step-size. We can specify the step size or let the solver choose the step size. Generally decreasing the step size increases the accuracy of the results while increasing the time required to simulate the system.

Variable-step solvers vary the step size during the simulation, reducing the step size to increase accuracy when a model's states are changing rapidly and increasing the step size to avoid taking unnecessary steps when the model's states are changing slowly. Computing the step size adds to the computational overhead at each step but can reduce the total number of steps, and hence simulation time, required to maintain a specified level of accuracy for models with rapidly changing or piecewise continuous states.

Continuous and Discrete Solvers

Continuous solvers use numerical integration to compute a model's continuous states at the current time step from the states at previous time steps and the state derivatives. Continuous solvers rely on the model's blocks to compute the values of the model's discrete states at each time step. Mathematicians have developed a wide variety of numerical integration techniques for solving the ordinary differential equations (ODEs) that represent the continuous states of dynamic systems. Simulink provides an extensive set of fixed-step and variable-step continuous solvers, each implementing a specific ODE solution method. Some continuous solvers subdivide the simulation time span into major and minor steps, where a minor time step represents a subdivision of the major time step. The solver produces a result at each major time step. It use results at the minor time steps to improve the accuracy of the result at the major time step.

Discrete solvers exist primarily to solve purely discrete models. They compute the next simulation time-step for a model and nothing else. They do not compute continuous states and they rely on the model's blocks to update the model's discrete states. We can use a continuous solver, but not a discrete solver, to solve a model that contains both continuous and discrete states. This is because a discrete solver does not handle continuous states. If you select a discrete solver for a continuous model, Simulink disregards your selection and uses a continuous solver instead when solving the model.

Simulink provides two discrete solvers, a fixed-step discrete solver and a variable-step discrete solver. The fixed-step solver by default chooses a step size and hence simulation rate fast enough to track state changes in the fastest block in our model. The variable-step solver adjusts the simulation step size to keep pace with the actual rate of discrete state changes in our model. This can avoid unnecessary steps and hence shorten simulation time for multirate models.

MODEL EXECUTION PHASE

In the simulation model execution phase, Simulink successively computes the states and outputs of the system at intervals from the simulation start time to the finish time, using information provided by the model. The successive time points at which the states and outputs are computed are called time steps. The length of time between steps is called the step size. The step size depends on the type of solver used to compute the system's continuous states, the system's fundamental sample time, and whether the system's continuous states have discontinuities (Zero Crossing Detection). At the start of the simulation, the model specifies the initial states and outputs of the system to be simulated. At each step, Simulink computes new values for the system's inputs, states, and outputs and updates the model to reflect the computed values. At the end of the simulation, the model reflects the final values of the system's inputs, states, and outputs. At each time step:

1) Simulink Updates the outputs of the models' blocks in sorted order. Simulink computes a block's outputs by invoking the block's output function. Simulink passes the current time and the block's inputs and states to the output function as it may require these arguments to compute the block's output. Simulink updates the output of a discrete block only if the current step is an integral multiple of the block's sample time.
2) Updates the states of the model's blocks in sorted order. Simulink computes a block's discrete states by invoking its discrete state update function. Simulink computes a block's

continuous states by numerically integrating the time derivatives of the continuous states. It computes the time derivatives of the states by invoking the block's continuous derivatives function.

3) Optionally checks for discontinuities in the continuous states of blocks. Simulink uses a technique called zero crossing detection to detect discontinuities in continuous states.
4) Computes the time for the next time step.

Simulink repeats steps 1 through 4 until the simulation stop time is reached.

Block Sorting Rules

Simulink uses the following basic update rules to sort the blocks:

1) Each block must be updated before any of the direct-feed through blocks that it drives. This rule ensures that the inputs to direct-feed through blocks will be valid when they are updated.
2) Non direct-feed through blocks can be updated in any order as long as they are updated before any direct-feed through blocks that they drive. This rule can be met by putting all non direct-feed through blocks at the head of the update list in any order. It thus allows Simulink to ignore non direct-feed through blocks during the sorting process.

The result of applying these rules is an update list in which non direct-feed through blocks appear at the head of the list in no particular order followed by direct-feed through blocks in the order required to supply valid inputs to the blocks they drive. During the sorting process, Simulink checks for and flags the occurrence of algebraic loops, that is, signal loops in which an output of a direct-feed through block is connected directly or indirectly to one of the block's inputs. Such loops seemingly create a deadlock condition since Simulink needs the input of a direct-feed through block in order to compute its output. However, an algebraic loop can represent a set of simultaneous algebraic equations (hence the name) where the block's input and output are the unknowns. Further, these equations can have valid solutions at each time step. Accordingly, Simulink assumes that loops involving direct-feed through blocks do, in fact, represent a solvable set of algebraic equations and attempts to solve them each time the block is updated during a simulation.

Parameter	Value
Rated power	3.02 kW
Rated torque	4.35 Nm
Rated speed	6724 rpm
Rated voltage	48 V
Rated current	75A
Ra	0.48 Ohm
La	1.4 mH
J	0.0117
B	0
Km	0.0631 Nm/A
Kg	1/138 V/rpm

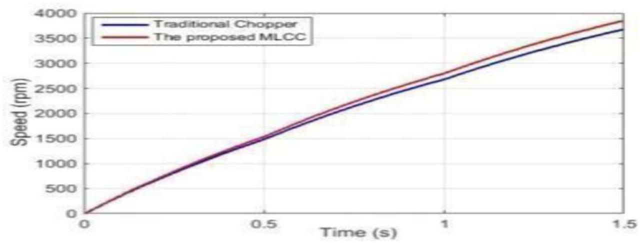


Table no.4.1

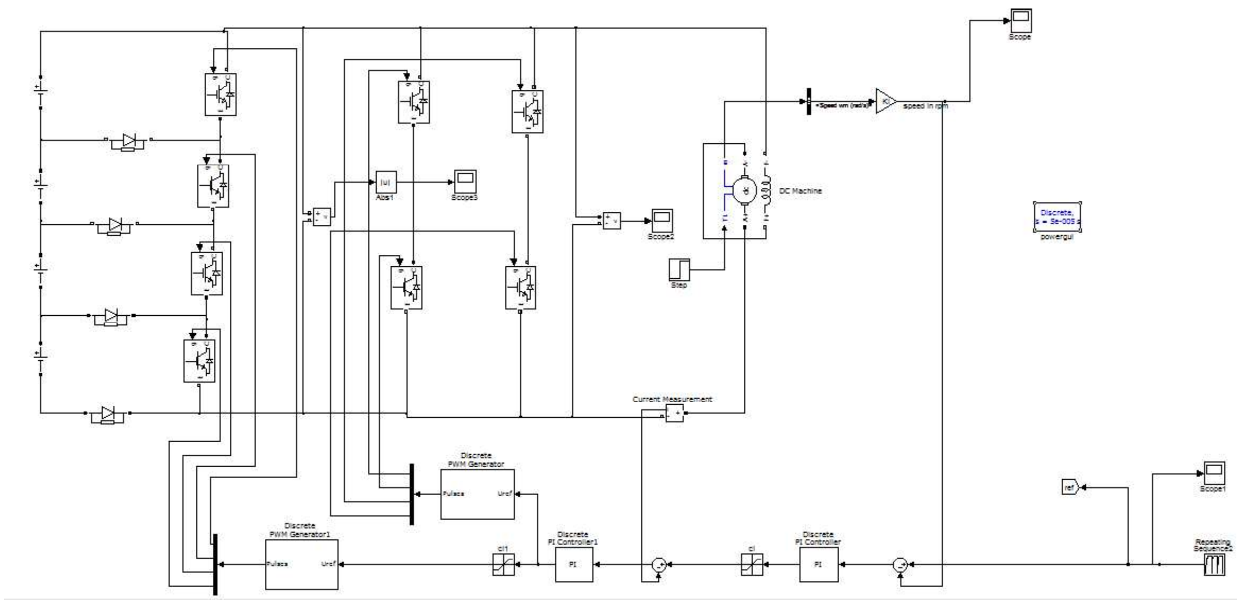


Fig 4.1 Proposed circuit configuration

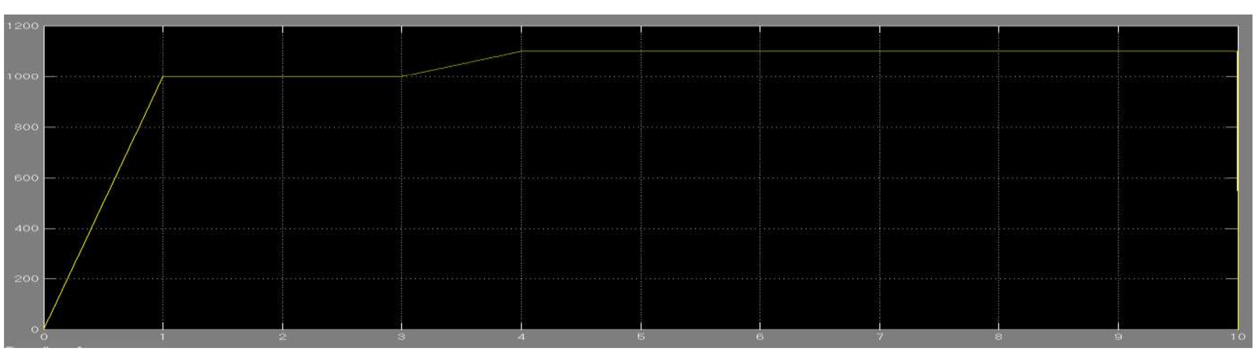
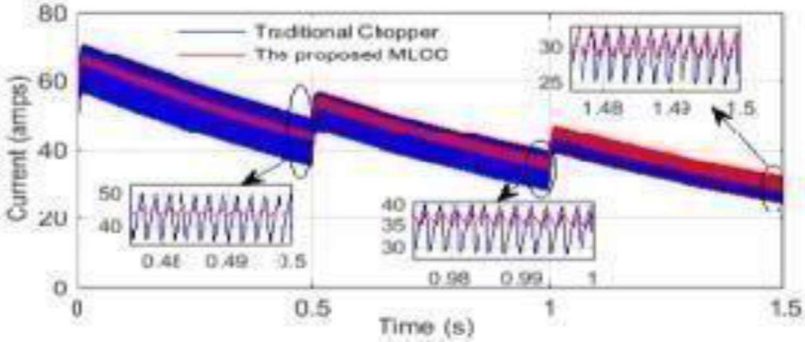


Fig 4.2 Reference speed



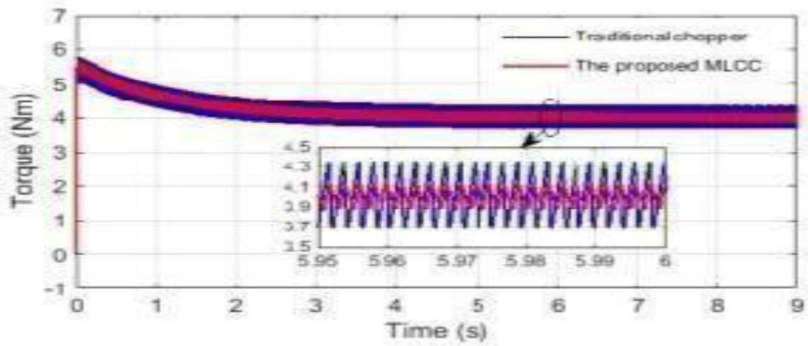


Fig.4.3 Comparison b/w traditional chopper and proposed MLCC

To evaluate the performance of the proposed multilevel chopper circuit, two Simulink models are built. One model is given for the proposed MLCC with the structure given in section II and the other one is given for the traditional chopper circuit with one fixed DC voltage source and one controllable switching element. The traditional chopper circuit works in a step-down mode to achieve the required reference voltage. The results of the output voltage performance is shown where the reference voltage (red) is changed from 40V to 20V then to 30V and thereafter to 6V for a duration of 0.05 s for each voltage level. It is clear from the comparison of these two figures that the proposed system works properly to provide the required level voltage to the load. The system keeps switching only between two consecutive voltage levels, While the tradition system of Fig must follow hard switching across the ultimate voltage range (0-48V) to provide the required reference voltage. The torque ripple in the steady-state performance is shown in the magnified part of Fig for the both drive systems.

The peak to peak torque ripple is approximately 0.7 Nm for the traditional chopper drive circuit and around 0.2 Nm for the proposed MLCC. It can be roughly concluded that with the proposed MLCC, the torque ripple is decreased by a factor of n (equal to the number of the DC voltage source cells). This reduction in the torque ripples eventually reflects on less noise and reduced mechanical vibration. In addition, the armature current in Fig which is like the developed torque performance, shows smaller current ripples for the proposed MLCC. These smaller ripples lead to less ohmic losses, and less harmonics and low EMI noise. The corresponding

speed problem in Fig.4.1 shows that for the same applied average voltage, the proposed MLCC gives a relatively higher speed level. In addition, the magnified part of the Fig.13 shows that the speed pulsation is higher in the case of the traditional chopper.

To accurately evaluate the proposed topology, the MLCC is simulated in closed loop control mode. The simulated circuit is arranged as given in Fig.1. The tuned parameters of the current loop controller are (Ki D 0:2 and Kp D 2, and high voltage limits of 48V), and that of the speed controller are (Ki D 16 and Kp D 1:6, and current limit of 90 amps). The speed reference is changed from initial value of 100 rad/s to 120 rad /s or (in rpm as 100_60/2_to 120_60/2_) at time tD 3 second. The load torque is changed from an initial value of 4 Nm to a intial value of 2 Nm at time t D 6 second. The detailed speed and torque performances are shown in Fig respectively. The controlled speed performance for the both methods (the proposed in red and the traditional in blue) shows almost the same general performance at starting and at steady state. However, the magnified parts of the traditional chopper show that the proposed MLCC has smallerspeed ripples and relatively smaller overshoot during the load change at time D 6 second. On the other hand, the corresponding torque profile shows an outstanding performance of the proposed MLCC as can be seen in the magnified parts. Although the average dynamic timeresponse is almost the same, the torque overshoot at load torque change point is higher than theproposed MLCC, in addition the torque ripples of the traditional chopper circuit are extremelyhigh.

CHAPTER 5

CONCLUSION

This project presents simulation results and experimental validation of a new topology of the multilevel chopper DC/DC converter for a DC motor system. The main objective of the propounded topology is to reduce current ripples and torque ripples that are associated with hard switching of the traditional chopper circuit. The proposed configuration provides constant -ve values of standard cell voltage and has the ability to generate the required nonstandard voltage within the cell voltage ranges. The generated voltage pattern of this topology has relatively smaller switching ripples compared to the traditional step-down DC/DC power converters. It has been shown that the operation of the DC motor with the new proposed chopper topology can efficiently decrease the motor armature current ripples and torque ripples by a factor equal to the number of the connected voltage cells. As compared with the operation of the motor with traditional chopper circuit.

CHAPTER 6

REFERENCES

[1] Hirofumi Akagi, Shin-ichi Kinouchi, and Yuji Miyazaki, "Bidirectional Isolated Dual – Active – Bridge (DAB) DC-DC Converters Using 1.2- KV 400A SiC MOSFET Dual Modules," CPSS Trans. on Power Electronics and Applications, vol. 1, no.1 pp. 33-40, December 2016.

[2] Lung-Sheng Yang and Tsorng-Juu Liang, 2012, "Analysis and Implementation of a Novel Bidirectional DC-DC Converter," IEEE Trans. on Industrial Electronics, vol. 59, no. 1, pp. 422-434, January 2012.

[3] Fan Zhang, Xu Zhang, Min Zhang, Abigail S. E. Edmonds, "Literature Review of Electric Vehicle Technology and its Applications," 5th Inter.Conf. on Computer Science and Network Technology (ICCSNT), 2016, pp. 832-837.

[4] Jorge Varela Barreras, Cl´audio Pinto, Ricardo de Castro, , Erik Schaltz, Søren Juhl Andreasen, Peter Omand Rasmussen, and Rui Esteves Ara´ujo, "Evaluation of a Novel BEV Concept Based on Fixed and Swappable Li-Ion Battery Packs," IEEE Trans. on Industry Applications, vol. 52, no. 6, November/December 2016.

[5] Dr. Rateb H. Issa, "Separately Excited DC Motor Optimal Efficiency Cotroller," International Journal of Engineering and Innovative Technology (IJEIT), Vol. 3, pp. 533-539, July 2013.

[6] Ubaid Bashir Qureshi and Shiekh Javaid Iqbal, "Development of Wind Turbine Simulator using Separately Excited DC Motor for Induction Generator Testing," IEEE 2016.

[7] K.Shreelekha and S.Arulmozhi, "Multiport Isolated Bidirectional DCDC Converter Interfacing Battery and Supercapacitor For Hybrid Energy Storage Application," In International

Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT), IEEE 2016, pp. 2763-2768.

[8] Udupi R. Prasanna, Anant Kumar Singh and Kaushik Rajashekara, "Novel bidirectional Single-phase Single-Stage Isolated AC-DC Converter With PFC for Charging of Electric Vehicles," IEEE Trans. on Transportation Electrification,vol. 3, no. 3, pp. 536-544, 2017.

[9] Amir Tokic, Admir Jukan, and Jasmin Smajic, 2016, "Parameter Estimation of Single-Phase Rectifier-Based Loads: Analytical Approach," IEEE Trans. on Power Delivery. IEEE Trans. on Power Delivery, vol. 31, pp. 532-540, April 2016.

[10] M. H. Rashid, Power Electronics: Circuits, Devices and Applications, 3rd Edition, 2004.

[11] Ali Emadi, Young Joo Lee and Kaushik Rajashekara, "Power Electronics and Motor Drives in Electric, Hybrid Electric, and Plug – In Hybrid Electric Vehicles," IEEE Trans. on industrial electronics, vol. 55, no. 6, june 2008.

[12] I. M. Oise, U. H. Diala, G. N. Ezeh and R. Opara, "Evaluation of Transient Response of a Separately Excited DC Motor under No-Load Condition," IEEE International Conference on Emerging & Sustainable Technologies for Power & ICT in a Developing Society (NIGERCON), 2013.

[13] Zhen Zhang, and Kwok-Tong Chau, "Pulse-Width–Modulation-Based Electromagnetic Interference Mitigation of Bidirectional Grid Connected Converters for Electric Vehicles," IEEE Trans. on Smart Grid, Vol. 8, No. 6, November 2017.

[14] Sivadas .M .C and Poorani Shivkumar, "Comprehensive Analyses and Comparison of 1kw Isolated DC-DC Converters For Bidirectional Ev Charging Systems," International Journal of Pure and Applied Mathematics, Vol. 118, no. 20, pp. 313-319, 2018.

[15] Malik, M.Z.; Chen, H.; Nazir, M.S.; Khan, I.A.; Abdalla, A.N.; Ali, A.; Chen,W. A New Ecient Step-Up Boost Converter with CLD Cell for Electric Vehicle and New Energy Systems. Energies **2020, 13, 1791. [CrossRef]**

[16] Nguyen, M.K.; Choi, Y.O. Voltage Multiplier Cell-Based Quasi-Switched Boost Inverter with Low Input Current Ripple. Electronics **2019, 8, 227. [CrossRef]**

[17] Ismail, A.A.A.; Elnady, A. Advanced Drive System for DC Motor Using Multilevel DC/DC Buck Converter Circuit. IEEE Access **2019, 7, 54167–54178. [CrossRef]**

[18] Lim, S.; Lee,H.; Cha, H.; Park, S. Multi-Level DC/DC Converter for E-Mobility Charging Stations. IEEE Access **2020, 8, 48774–48783. [CrossRef]**

[19] Dimitrov, B.; Hayatleh, K.; Barker, S.; Collier, G.; Sharkh, S.; Cruden, A. A Buck-Boost Transformerless DC–DC Converter Based on IGBT Modules for Fast Charge of Electric Vehicles. Electronics **2020, 9, 397.** [CrossRef]

APPENDIX

MATLAB is a high-performance language for technical computing. It integrates computation, visualization, and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation. Typical uses include-

- Math and computation
- Algorithm development
- Data acquisition
- Modeling, simulation, and prototyping
- Data analysis, exploration, and visualization
- Scientific and engineering graphics

MATLAB is an interactive system whose basic data element is an array that does not require dimensioning. This allows solving many technical computing problems, especially those with matrix and vector formulations, in a fraction of the time it would take to write a program in a scalar non-interactive language such as C or FORTRAN.

The MATLAB system consists of six main parts:

(a) Development Environment

This is the set of tools and facilities that help to use MATLAB functions and files. Many of these tools are graphical user interfaces. It includes the MATLAB desktop and Command Window, a command history, an editor and debugger, and browsers for viewing help, the workspace, files and the search path.

(b) The MATLAB Mathematical Function Library

This is a vast collection of computational algorithms ranging from elementary functions, like sum, sine, cosine, and complex arithmetic, to more sophisticated functions like matrix inverse, matrix Eigen values, Bessel functions, and fast Fourier transforms.

(c) The MATLAB Language

This is a high-level matrix/array language with control flow statements, functions, data structures, input/output, and object-oriented programming features. It allows both "programming in the small" to rapidly create quick and dirty throw-away programs, and "programming in the large" to create large and complex application programs.

(d) Graphics

MATLAB has extensive facilities for displaying vectors and matrices as graphs, as well as annotating and printing these graphs. It includes high-level functions for two-dimensional and three-dimensional data visualization, image processing, animation, and presentation graphics. It also includes low-level functions that allow to fully customize the appearance of graphics as well as to build complete graphical user interfaces on MATLAB applications.

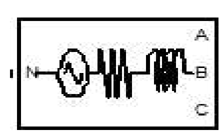
(e) The MATLAB Application Program Interface (API)

This is a library that allows writing in C and FORTRAN programs that interact with MATLAB. It includes facilities for calling routines from MATLAB (dynamic linking), calling MATLAB as a computational engine, and for reading and writing MAT-files.

(f) MATLAB Documentation

MATLAB provides extensive documentation, in both printed and online format, to help to learn about and use all of its features. It covers all the primary MATLAB features at a high level, including many examples. The MATLAB online help provides task-oriented and reference information about MATLAB features. MATLAB documentation is also available in printed form and in PDF format.

(1) Three phase source block

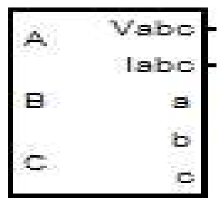


Three Phase Source Block

The Three-Phase Source block implements a balanced three-phase voltage source with internal R-L impedance. The three voltage sources are connected in Y with a neutral connection that can be internally ground.

(2) VI measurement block

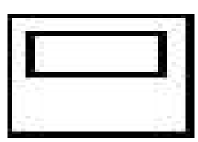
The Three-Phase V-I Measurement block is used to measure three-phase voltages and currents in a circuit. When connected in series with three-phase elements, it returns the three phase-to-ground or phase-to-phase voltages and the three line currents



Three Phase V-I Measurement

(3) Scope

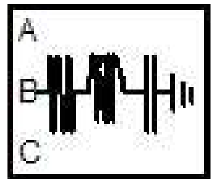
Display signals generated during a simulation. The Scope block displays its input with respect to simulation time. The Scope block can have multiple axes (one per port); all axes have a common time range with independent y-axes. The Scope allows you to adjust the amount of time and the range of input values displayed. You can move and resize the Scope window and you can modify the Scope's parameter values during the simulation



Scope

(4) Three-Phase Series RLC Load

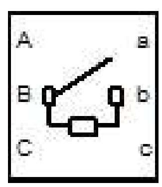
The Three-Phase Series RLC Load block implements a three-phase balanced load as a series combination of RLC elements. At the specified frequency, the load exhibits constant impedance. The active and reactive powers absorbed by the load are proportional to the square of the applied voltage.



Three-Phase Series RLC Load

(5) Three-Phase Breaker block

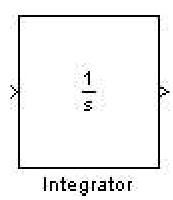
The Three-Phase Breaker block implements a three-phase circuit breaker where the opening and closing times can be controlled either from an external Simulink signal or from an internal control signal.



Three-Phase Breaker Block

(6) Integrator

Library: Continuous

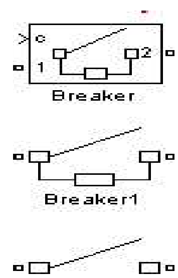


Integrator

The integrator block outputs the integral of its input at the current time step. The following equation represents the output of the block y as a function of its input u and an initial condition y_0, where y and u are vector functions of the current simulation time t.

(7) Breaker : Implement circuit breaker opening at current zero crossing.

Library: Elements



Circuit Breaker

Purpose: The Breaker block implements a circuit breaker where the opening and closing times can be controlled either from an external SIMULINK signal (external control mode), or from an internal control timer (internal control mode).

A series R_s-C_s snubber circuit is included in the model. It can be connected to the circuit breaker. If the Breaker block happens to be in series with an inductive circuit, an open circuit or a current source, you must use a snubber.

When the breaker block is set in external control mode, a SIMULINK input appears on the block icon. The control signal connected to the SIMULINK input must be either 0 or 1 (0 to open the breaker, 1 to close it).

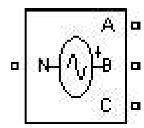
When the Breaker block is set in internal control mode, the switching times are specified in the dialog box of the block.

When the breaker is closed, it is represented by a resistance R_{on}. The Ron value can be set as small as necessary in order to be negligible compared with external components (a typical value is 10 m ohms). When the breaker is open, it has an infinite resistance.

(8) Three-Phase Programmable Voltage Source

Implement three-phase voltage source with programmable time variation of amplitude, phase, frequency, and harmonics

Library: Electrical Sources



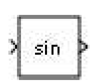
Three Phase Voltage Sources

Purpose: This block is used to generate a three-phase sinusoidal voltage with time-varying parameters. It can be programmed with the time variation for the amplitude, phase or frequency of the fundamental component of the source. In addition, two harmonics can be programmed and superimposed on the fundamental signal.

(9) Trigonometric Function

Specified trigonometric function on input

Library: Math Operations



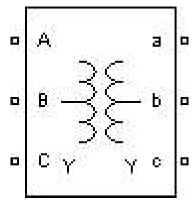
Trigonometric Function

Purpose: The Trigonometric Function block performs common trigonometric functions

(10) Three-Phase Transformer (Two Windings)

Implement three-phase transformer with configurable winding connections

Library: **Elements**



Three Phase Transformer

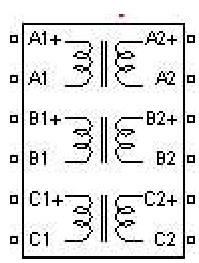
Purpose:

The Three-Phase Transformer (Two Windings) block implements a three-phase transformer using three single-phase transformers. The saturation characteristic, when activated, is the same as the one described for the saturable Transformer block, and the icon of the block is automatically updated. If the fluxes are not specified, the initial values are automatically adjusted so that the simulation starts in steady state.

(11) Three-Phase Transformer 12 Terminals

Implement three single-phase, two-winding transformers where all terminals are accessible

Library: Elements



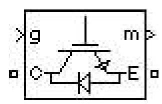
Two winding Transformer

Purpose: The Three-Phase Transformer 12 Terminals block implements three single-phase, two-winding linear transformers where all the twelve winding connectors are accessible. The block can be used in place of the Three-Phase Transformer (Two Windings) block to implement a three-phase transformer when primary and secondary are not necessarily connected in Star or Delta.

(12) IGBT/Diode

Implements ideal IGBT, GTO, or MOSFET and antiparallel diode

Library: **Power Electronics**



IGBT

Purpose: The IGBT/Diode block is a simplified mode of an IGBT (or GTO or MOSFET)/Diode pair where the forward voltages of the forced-commutated device and diode are ignored.