MODELLING OF DC MOTOR USING SIMULINK

511946. LOKESH LOYA

ABSTRACT

DC motor acts

As an energy conversion actuator that converts electrical energy (of source) into mechanical energy (for load). These motors are extensively applied for robotic manipulations, cutting tools, electrical tractions, etc. The torque-speed characteristics of DC motors are most compatible with most mechanical loads.

Many industrial applications require high-performance rotating electric drives. A proposed DC drive has precise speed control, stable operation in a complete range of speeds, and good transient behavior with smooth and step-less control. In this paper, the modeling of a DC motor is performed by using a generalized equation in MATLAB.

In the present work transfer function and Modeling of a DC motor are performed by using generalized equations in MATLAB and Simulink.

INTRODUCTION SIMULINK

A common actuator in control systems is the DC motor. It directly provides rotary motion and, coupled with wheels or drums and cables can provide translational motion to Electric Vehicles. The electric equivalent circuit of the armature and the free-body diagram of the rotor are shown in the following figure.

The input of the system is the voltage source (V) applied to the motor's armature, while the output is the rotational speed of the shaft. The rotor and shaft are assumed to be rigid.

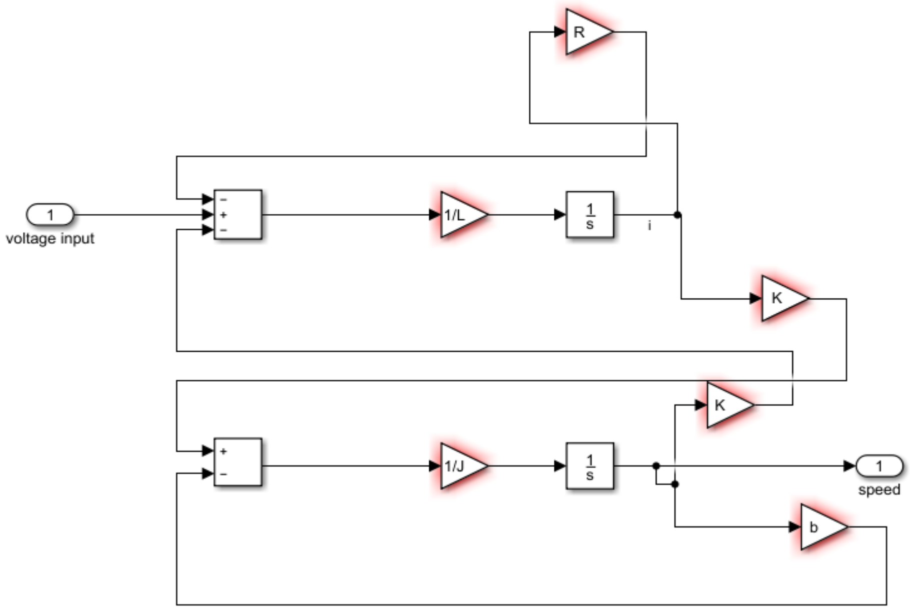
A DC motor is an electromechanical energy converter that converts electrical energy into mechanical energy. It is often used as an actuator in control systems.

The DC motor acts as an energy conversion actuator that converts electrical energy (of source) into mechanical energy. (for load). These motors are extensively applied for robotic manipulations, cutting tools, electrical tractions, etc. The torque-speed characteristics of DC motors are most compatible with most mechanical loads.

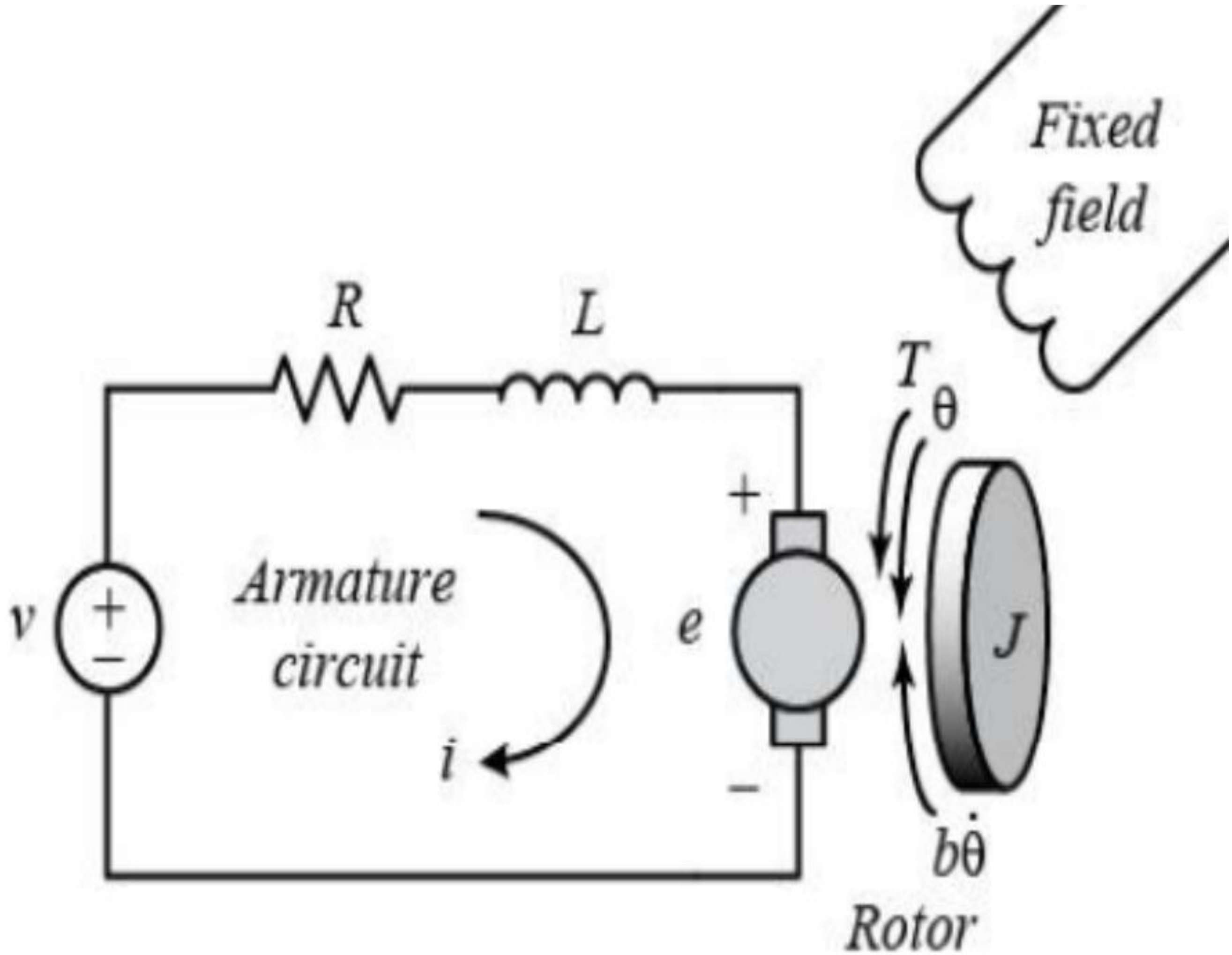
Hence DC motors are always a good ground for advanced control algorithms. The control characteristics of these motors have resulted in their immense use and hence control of their speed is required. The speed of a DC motor depends on supply voltage, armature resistance, and field flux produced by the field current. The methods to control the speed of these motors are armature voltage control, armature resistance control, and field flux control.

II. MODELING AND BLOCK DIAGRAMS OF A DC MOTOR

As The DC motor model is implemented by mathematical. expression. This implemented model is analyzed by techniques of time domain and frequency domain analysis in MATLAB *simulation.*



Block diagram of DC motor



Free body diagram of armature of a DC motor

The basic equations of a DC motor (electric part) are obtained from Maxwell's electromagnetic theory. For this example, we will assume that the input of the system is the voltage source (V) applied to the motor's armature, while the output is the rotational speed of the shaft d(theta)/dt. The rotor and shaft are assumed to be rigid. We further assume a viscous friction model, that is, the friction torque is proportional to shaft angular velocity.

For continuous linear time-invariant (LTI) systems, the standard state-space representation is given below:

$$\bar{x}' = Ax + Bu$$

$$y = C\bar{x} + Du$$

where x is the vector of state variables (nx1), X is the time derivative of the state vector (nx1), u is the input or control vector (px), y is the output vector (qx1), A is the system matrix (nxn), B is the input matrix (nxp), C is the output matrix (in), D is the feedforward matrix (qxp).

The output equation is necessary because often there are state variables that are not directly observed or are otherwise not of interest. The output matrix, C, is used to specify which state variables (or combinations thereof) are available for use by the controller. Also often there is no direct feedforward in which case D is the zero matrix

In general, the torque generated by a DC motor is proportional to the armature current and the strength of the magnetic field. In this example, we will assume that the magnetic field is constant and, therefore, that the motor torque is proportional to only the armature current I by a constant factor Kt as shown in the equation below. This is referred to as an armature-controlled motor

$$T = Ki$$

The back emf, e, is proportional to the angular velocity of the shaft by a constant factor Ke

$$e = K_e\theta$$

In SI units, the motor torque and back emf constants are equal, that is, Kt = Ke; therefore, we will use K to represent *both the motor torque constant and the back emf constant.*

From the figure above, we can derive the following governing equations based on Newton's 2nd law and Kirchhoff's voltage law

JÖ+b0= J0+b 0= Ki

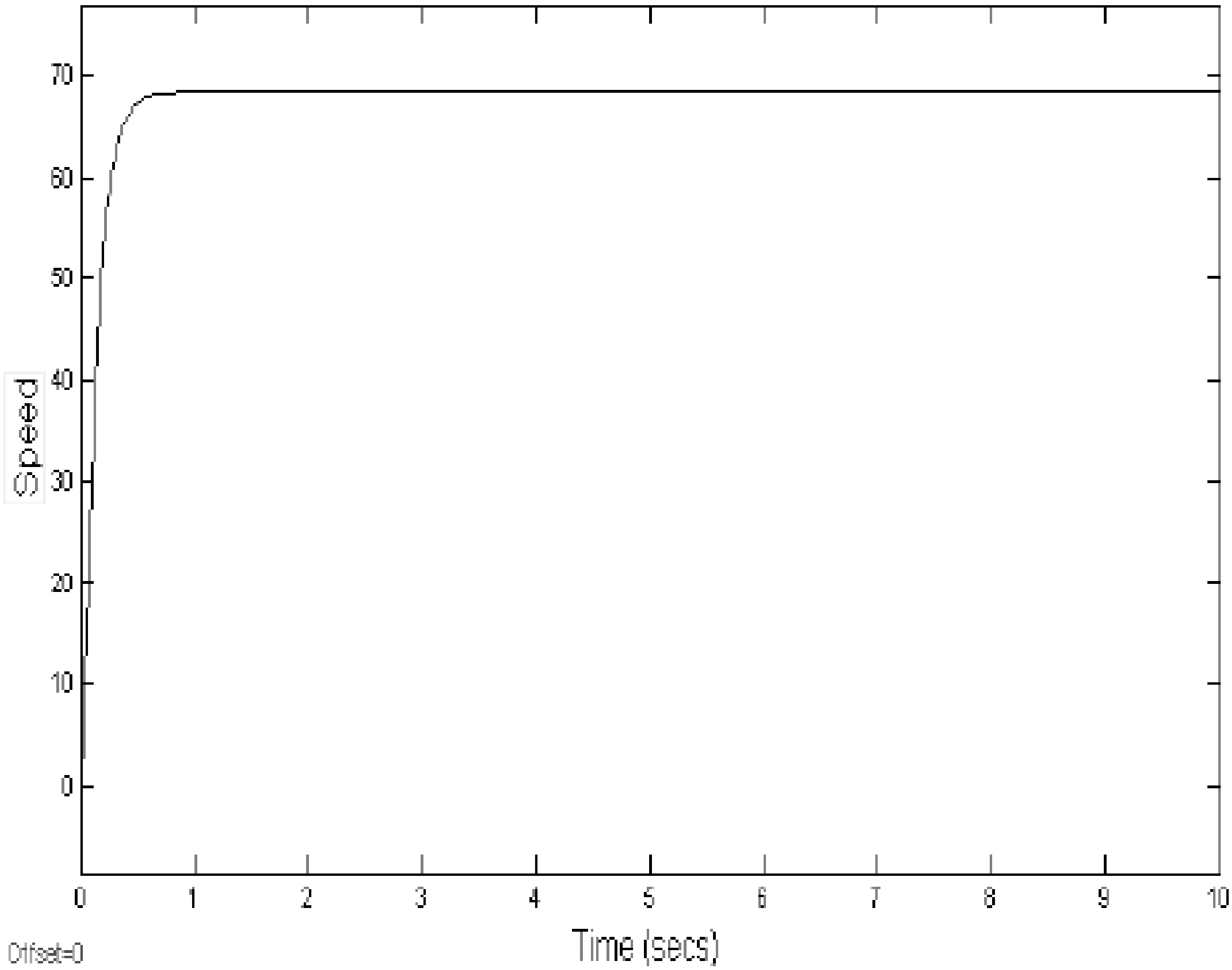
di /dt L +Ri=V-KO

where J stands for a moment of inertia, b stands for friction coefficient, L is armature inductance, R stands for armature resistance, V stands for input voltage, K stands for electromotive force constant, stands for angular acceleration.

In state-space form, the governing equations above can be expressed by choosing the rotational speed and electric current as of the state variables. Again the armature voltage is treated as the input and the rotational speed is chosen as the output.

STEP RESPONSE OF A DC MOTOR:

The rotational speed of a ***DC motor*** *is directly proportional to the mean (average) value of its supply voltage which is applied to the motor terminals.*



Physical Parameters

MOMENT OF INERTIA:

A quantity expressing a body's tendency to resist angular acceleration, which is the sum of the products of the mass of each particle in the body with the square of its distance from the axis of rotation

(J) moment of inertia of the rotor ->0.01 kg.m^2

Viscous friction constant:

*The ratio of the shearing stress to the velocity gradient is a measure of the viscosity of the fluid and is called the coefficient of viscosity η, or **η=Fx/Av**. The cgs unit for measuring the coefficient of viscosity is the poise. ... Viscosity is the principal factor resisting motion in laminar flow.*

(b) motor viscous friction constanT->0.1 N.m.s

BACK ELECTROMOTIVE FORCE CONSTANT (=BACK EMF CONSTANT):

Back electromotive force is the voltage generated at both ends of a winding due to changes in flux linkage in a shaft motor operation. Since the back electromotive force is proportional to the moving speed, it is a constant. The unit is [V/m/s].

(ke) electromotive force constant->0.01 V/rad/sec

TORQUE CONSTANT:

*The Torque Constant defines the torque-current relationship of a motor and is in Nm/amp**.***

The motor and torque constants are related by the formula Km = Kt(trap)/sqrt(R), where R is the phase-to-phase resistance of the winding.

(kt) motor torque constant->0.01 N.m/Amp

ELECTRIC RESISTANCE:

Resistance is a measure of the opposition to current flow in an electrical circuit. Resistance is measured in ohms, symbolized by the Greek letter omega (Ω).

(R) electric resistance->1 Ohm

ELECTRIC INDUCTANCE:

The property of an electric conductor or circuit that causes an electromotive force to be generated by a change in the current flowing. Inductance is measured in henry(H).

(L) electric inductance->0.5 H

TRANSFER FUNCTIONS

Transfer Function: Applying the Laplace transform, the above modeling equations can be expressed in terms of the Laplace variables.

The transfer function of a system is defined as the ratio of the Laplace transform of output to the Laplace transform of input where all the initial conditions are zero.

*Where, **T(S) = Transfer function** of the system. C(S) = output.*

T(S) =C(S)/R(S)

SYSTEM EQUATIONS:

In general, the torque generated by a DC motor is proportional to the armature current and the strength of the magnetic field. In this example we will assume that the magnetic field is constant and, *therefore, that the motor torque is proportional to only the armature current I by a constant factor Kas showed in the equation below. This is referred to as an armature-controlled motor.*

$$T = K_t i$$

The back emf, e is proportional to the angular velocity of the shaft by a constant factor Ke

$$e = K_e \theta$$

In SI units, the motor torque and back emf constants are equal, that is, $K_1 = Ke$; therefore, we will use it to represent both the motor torque constant and the back emf constant.

From the figure above, we can derive the following governing equations based on Newton's 2nd law and Kirchhoff's voltage law.

$$J\ddot{O}+b0= J0+b\ 0= Ki$$

$$di\ dt\ L +Ri=V-KO$$

Transfer function

$$s(Js+b)e(s) = KI(s)$$

$$(Ls + R)I(s) = V(s) – Kse(s)$$

We arrive at the following open-loop transfer function by eliminating I(s) between the two above equations, where the rotational speed is considered the output and the armature voltage is considered the input.

$$P(s)= \theta (s)/V (s)=K/(Js + b)(Ls + R),+ K^2 \text{ [rad/sec/v]}$$

3. Mathematical equations for DC motor.

$$J \theta''=T-b\theta' \Rightarrow \theta''=1(K_t i- b\theta')/J \quad \text{-------} (1)$$

$$Li'=-Ri+V \Rightarrow i'=(-Ri+V- K_e\theta')/L \quad \text{-------} (2)$$

MATLAB:

MATLAB was used to effectively analyze the suspension durability of an in-wheel SRM of an EV in comparison with an internal combustion engine (ICE) vehicle. This gave the starting point for the present study, as the subsequent findings can be placed in the context of those previously reported by other authors working on similar issues. Based on the studies identified during the above literature review, the same approach was adopted in the present study. Due to physical changes affected by the unsprung mass with the new in-wheel SRM, the vehicle response was also expected to change, as the vehicle suspension system remained unchanged.

Thus, this analysis aims to evaluate the changes due to the new in-wheel SRM. As part of this process, a small car was tested under different road conditions, with increased mass (due to *the motor) added to the wheel, reproducing the effects of the in-wheel SRM unsprung mass. A flow chart as shown in Figure 2 will explain the methodology for the study. The experimental results (load history curve) with the current new design have been obtained for fatigue analysis of the suspension to study the life cycle analysis of the new design and have been reported in Kulkarni et al.1 The study on tire/rim road interface has been reported in Kulkarni and Kapoor.23*

To evaluate the suspension performance due to increased mass, the transient response for frequency and Bode plot analyses were conducted using the MATLAB program to solve the quarter-car equations for two different cases to study the changes brought in due to the new design and were compared with the ICE vehicle. The sprung/unsprung / driver's seat mass studies for EVs with an in-wheel SRM and associated comparisons with ICE conducted in this research are novel for SRM EVs.

MATLAB code:

```
clc;
close all;
J=0.01;
b=0.1;
K=0.01;
R=1;
L=0.5;
s=tf('s');
P_motor = K/((J*s+b)*(L*s+R)+K^2);
linearSystemAnalyzer('step', P_motor, 0:0.1:5);
```

Transfer function:

```
P_motor =

          0.01
  ---------------------------
  0.005 s^2 + 0.06 s + 0.1001
```

```
Continuous-time transfer function.
```

Simulink:

In designing suitable isolators to reduce unwanted vibration in vehicles, the response from a mathematical model which characterizes the transmissibility ratio of the input and output of the vehicle is required.

In this study, a Matlab Simulink model is developed to study the dynamic behavior performance of a passive suspension system for a lightweight electric vehicle. The Simulink model is based on the two degrees of freedom system quarter car model.

The model is compared to the theoretical plots of the transmissibility ratios between the amplitudes of the displacements and accelerations of the sprung and unsprung masses to the amplitudes of the ground, against the frequencies at different damping values. It was found that the frequency responses obtained from the theoretical calculations and the Simulink simulation are comparable to each other. Hence, the model may be extended to a full vehicle model.

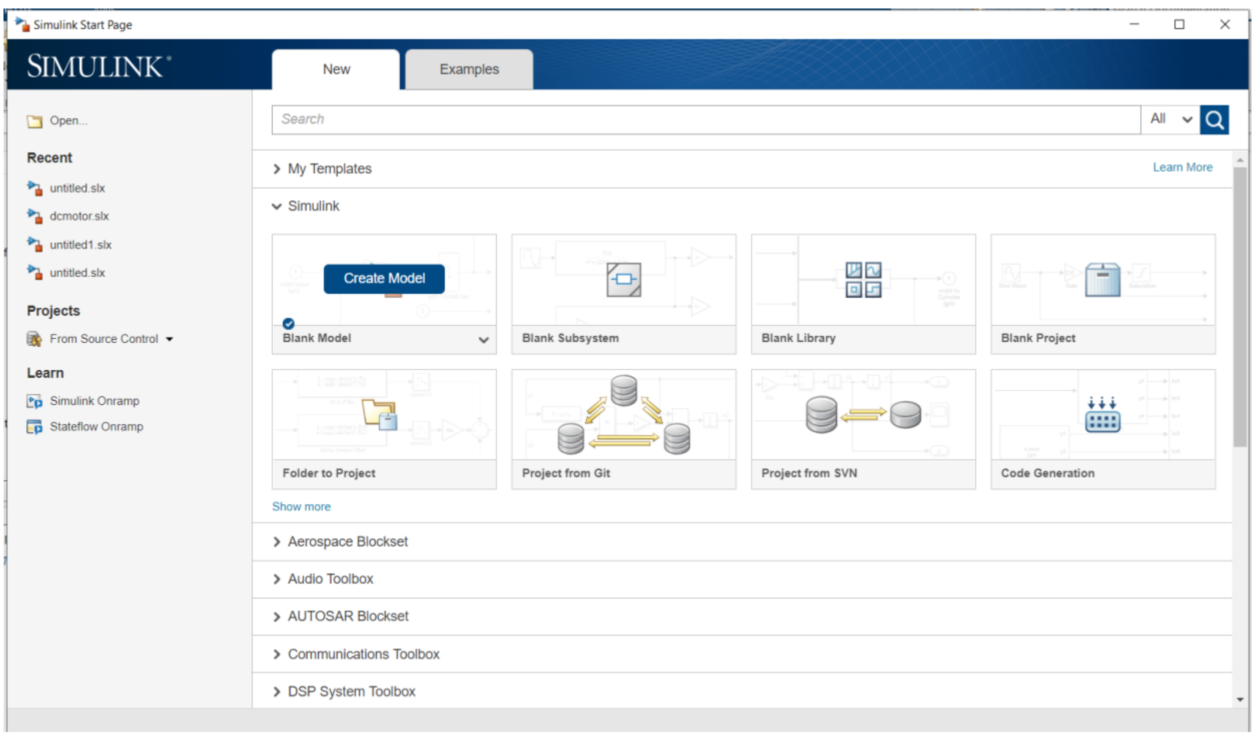
Simulink is a block diagram environment for multi-domain simulation and model-based design. Each block implements some function on its inputs and produces the results. The blocks are chosen and connected based on the mathematical equations derived for the model of the vehicle. A Simulink block diagram for the quarter vehicle model has been constructed based on the equation

• *Used to model, analyze and simulate dynamic systems using block diagrams.*

• *Simulink is a graphical, "drag and drop" environment for building simple and complex signal and system dynamic simulations - therefore is easy to use.*

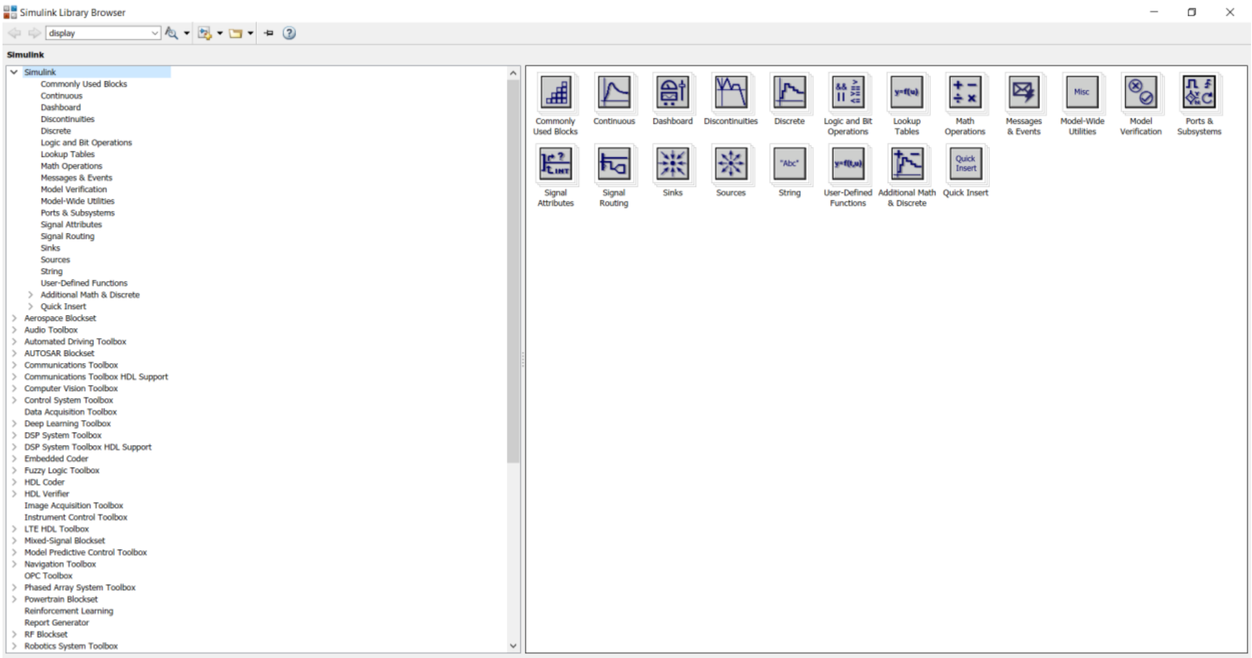
• *It allows users to concentrate on the structure of the problem, rather than having to worry about a programming language.*

Newmodel:



Click the new-model icon in the upper left corner to start a new Simulink file

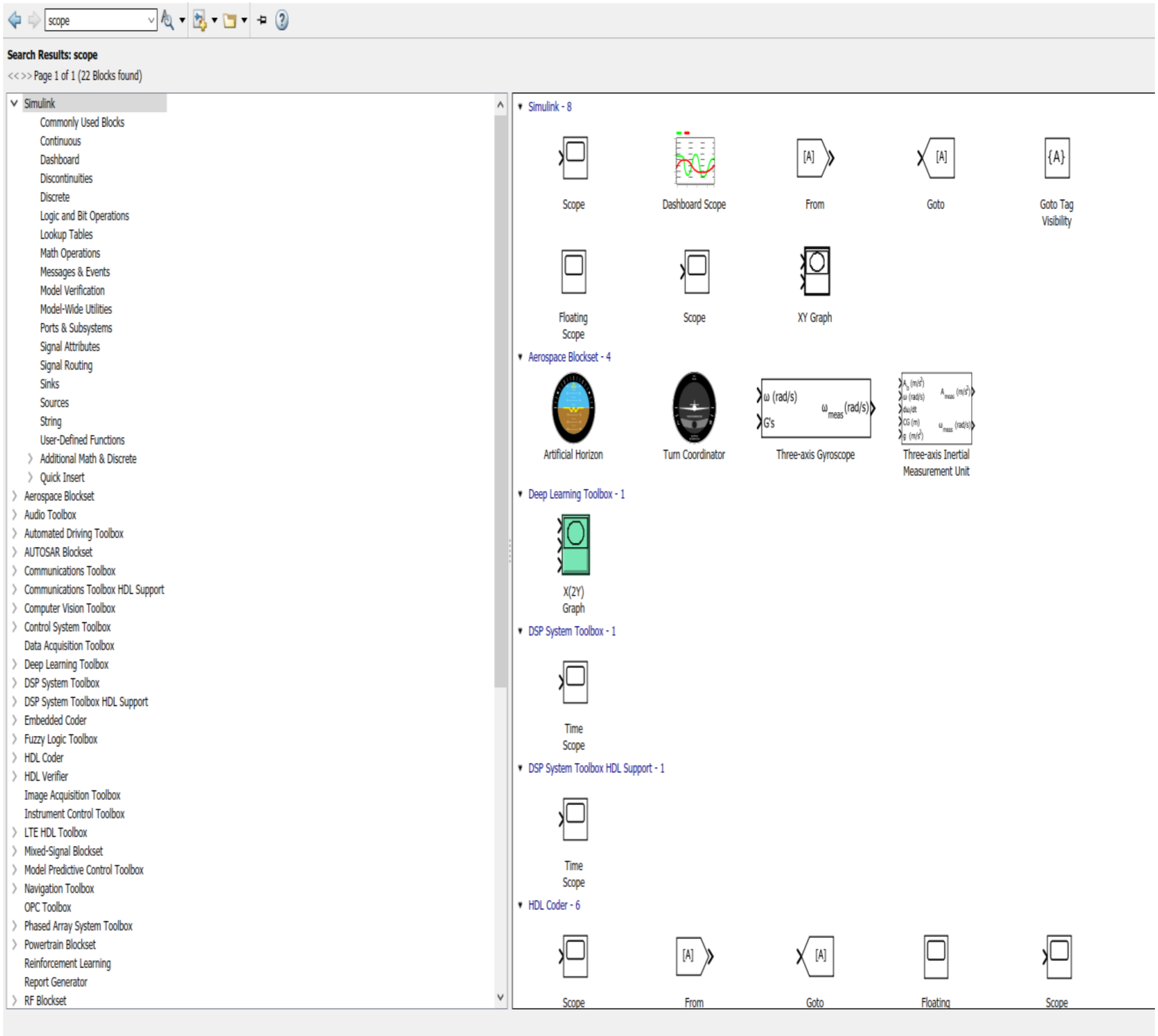
Create a new model under 'file/new/model'.

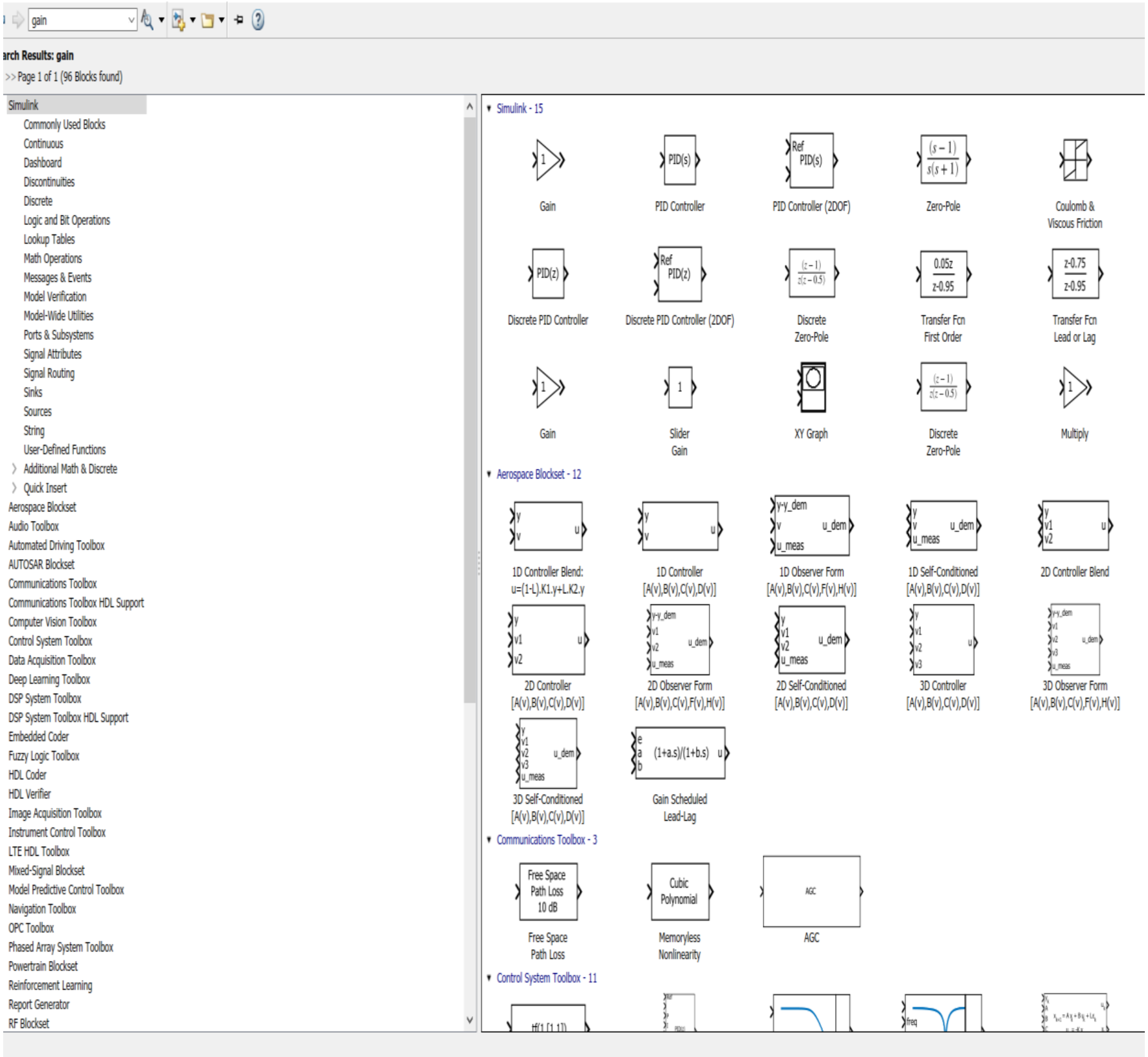


LIBRARIES:

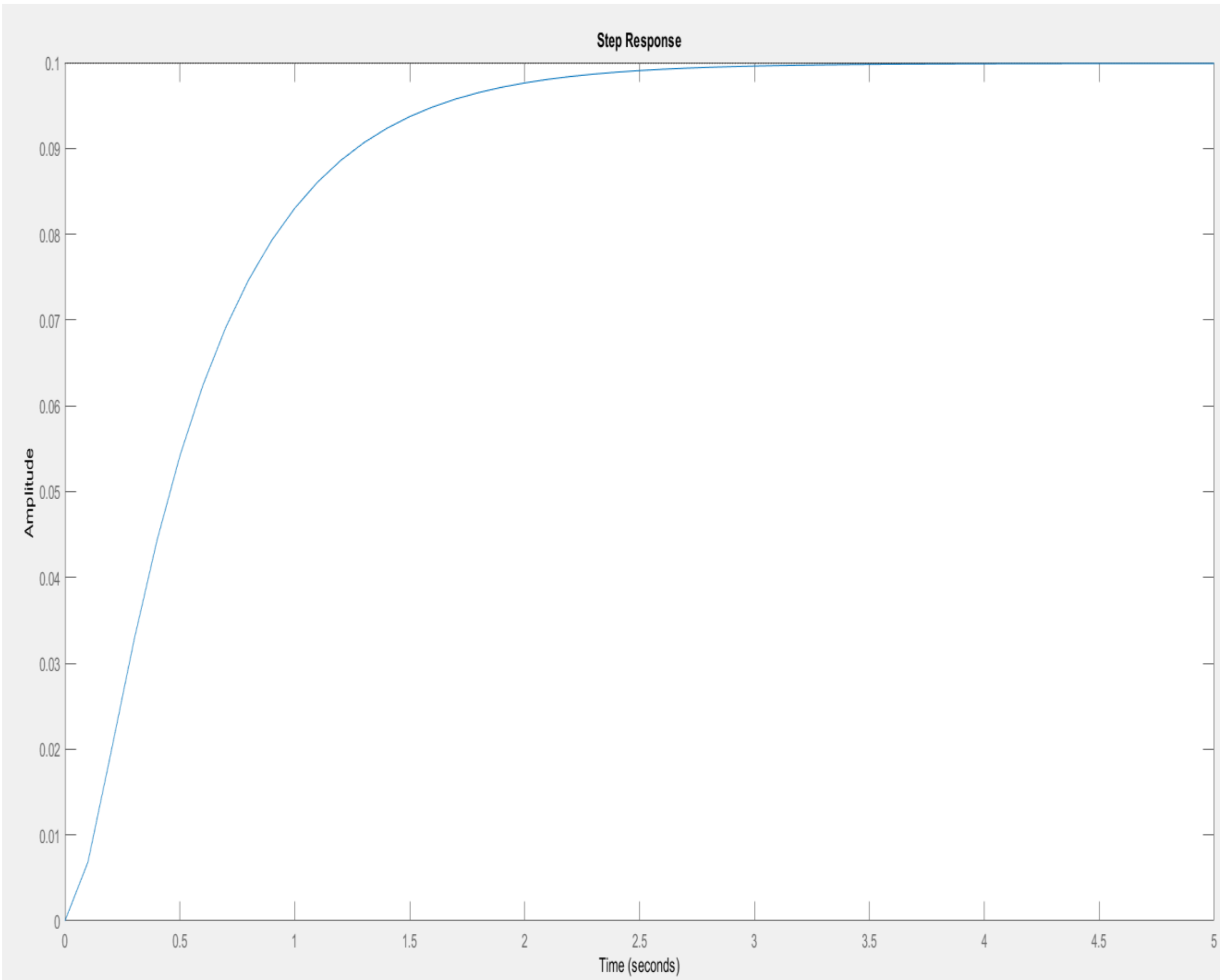
What are libraries in Matlab?

*A block library is **a collection of blocks that you can use in a Simulink® model**. You can create instances of blocks from built-in Simulink libraries, and you can make custom libraries for instances of blocks that you create. You can access the built-in libraries from the Simulink Library Browser.*

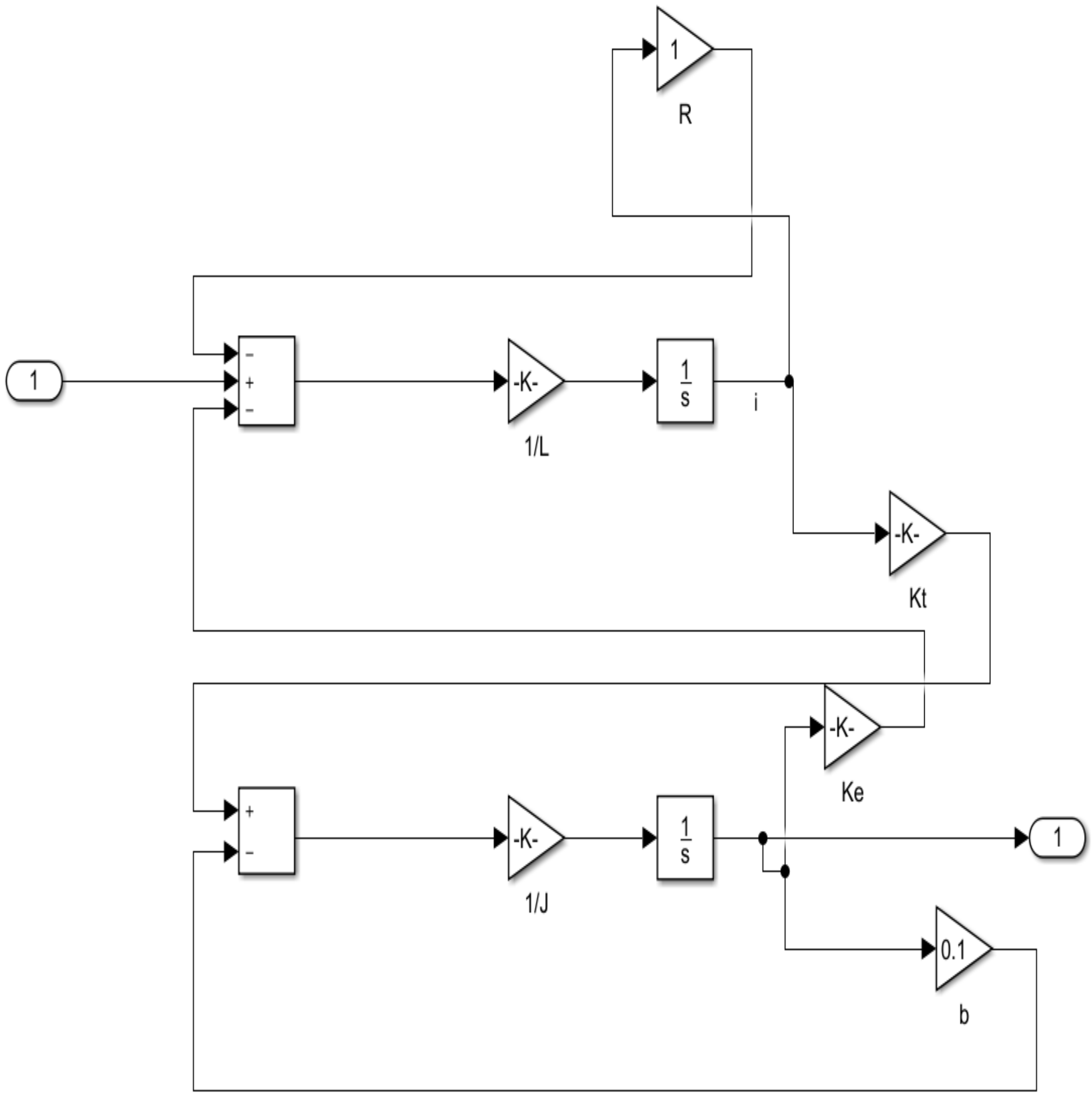

scope
Search Results: scope
<<>> Page 1 of 1 (22 Blocks found)
Simulink
Commonly Used Blocks
Continuous
Dashboard
Discontinuities
Discrete
Logic and Bit Operations
Lookup Tables
Math Operations
Messages & Events
Model Verification
Model-Wide Utilities
Ports & Subsystems
Signal Attributes
Signal Routing
Sinks
Sources
String
User-Defined Functions
Additional Math & Discrete
Quick Insert
Aerospace Blockset
Audio Toolbox
Automated Driving Toolbox
AUTOSAR Blockset
Communications Toolbox
Communications Toolbox HDL Support
Computer Vision Toolbox
Control System Toolbox
Data Acquisition Toolbox
Deep Learning Toolbox
DSP System Toolbox
DSP System Toolbox HDL Support
Embedded Coder
Fuzzy Logic Toolbox
HDL Coder
HDL Verifier
Image Acquisition Toolbox
Instrument Control Toolbox
LTE HDL Toolbox
Mixed-Signal Blockset
Model Predictive Control Toolbox
Navigation Toolbox
OPC Toolbox
Phased Array System Toolbox
Powertrain Blockset
Reinforcement Learning
Report Generator
RF Blockset
Simulink - 8
Scope
Dashboard Scope
From
Goto
Goto Tag Visibility
Floating Scope
Scope
XY Graph
Aerospace Blockset - 4
Artificial Horizon
Turn Coordinator
Three-axis Gyroscope
Three-axis Inertial Measurement Unit
Deep Learning Toolbox - 1
X(2Y) Graph
DSP System Toolbox - 1
Time Scope
DSP System Toolbox HDL Support - 1
Time Scope
HDL Coder - 6
Scope
From
Goto
Floating
Scope

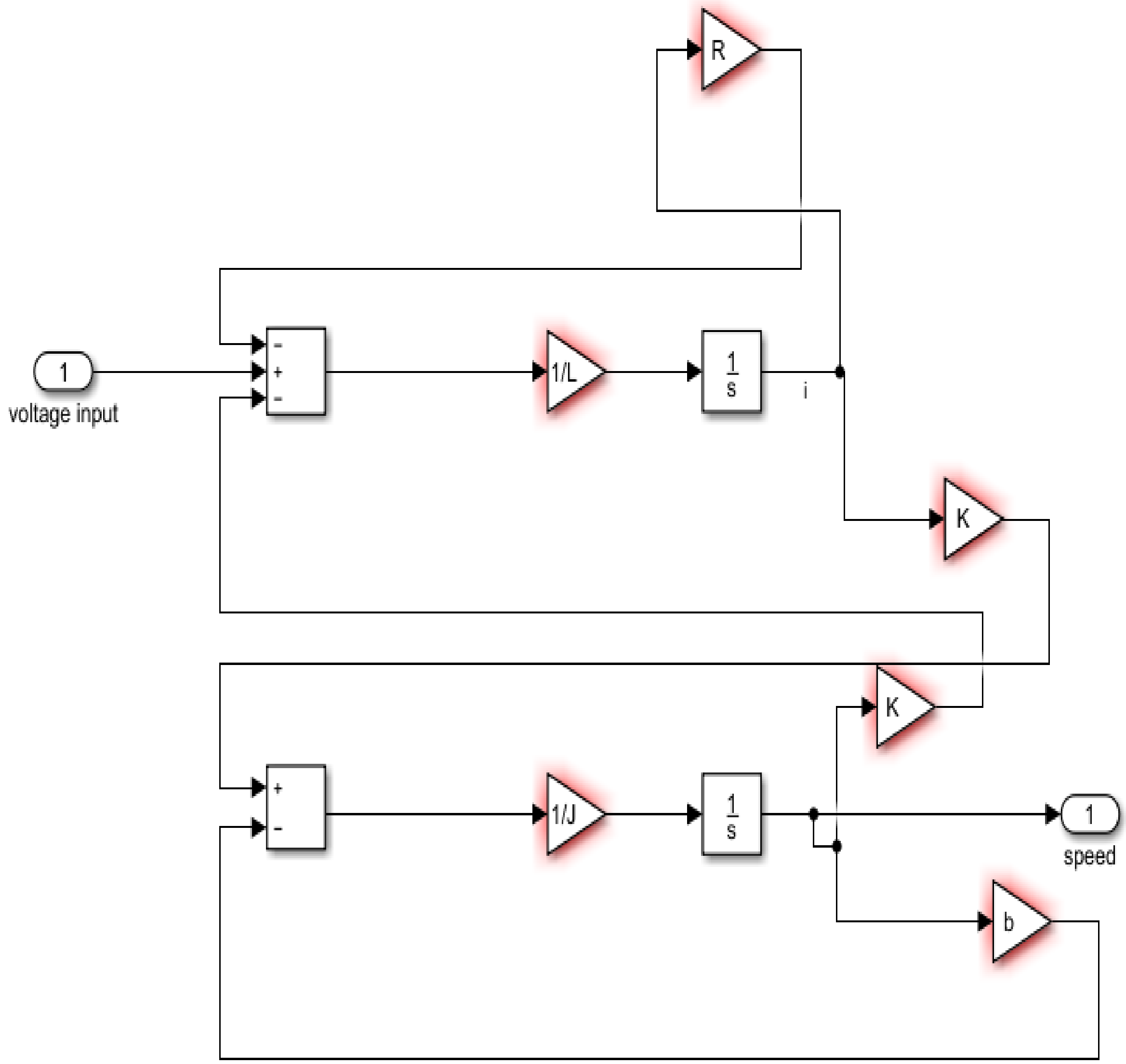

gain
arch Results: gain
>> Page 1 of 1 (96 Blocks found)
Simulink
Commonly Used Blocks
Continuous
Dashboard
Discontinuities
Discrete
Logic and Bit Operations
Lookup Tables
Math Operations
Messages & Events
Model Verification
Model-Wide Utilities
Ports & Subsystems
Signal Attributes
Signal Routing
Sinks
Sources
String
User-Defined Functions
Additional Math & Discrete
Quick Insert
Aerospace Blockset
Audio Toolbox
Automated Driving Toolbox
AUTOSAR Blockset
Communications Toolbox
Communications Toolbox HDL Support
Computer Vision Toolbox
Control System Toolbox
Data Acquisition Toolbox
Deep Learning Toolbox
DSP System Toolbox
DSP System Toolbox HDL Support
Embedded Coder
Fuzzy Logic Toolbox
HDL Coder
HDL Verifier
Image Acquisition Toolbox
Instrument Control Toolbox
LTE HDL Toolbox
Mixed-Signal Blockset
Model Predictive Control Toolbox
Navigation Toolbox
OPC Toolbox
Phased Array System Toolbox
Powertrain Blockset
Reinforcement Learning
Report Generator
RF Blockset
Simulink - 15
Gain
PID Controller
PID Controller (2DOF)
Zero-Pole
Coulomb & Viscous Friction
Discrete PID Controller
Discrete PID Controller (2DOF)
Discrete Zero-Pole
Transfer Fcn First Order
Transfer Fcn Lead or Lag
Gain
Slider Gain
XY Graph
Discrete Zero-Pole
Multiply
Aerospace Blockset - 12
1D Controller Blend: u=(1-L).K1.y+L.K2.y
1D Controller [A(v),B(v),C(v),D(v)]
1D Observer Form [A(v),B(v),C(v),F(v),H(v)]
1D Self-Conditioned [A(v),B(v),C(v),D(v)]
2D Controller Blend
2D Controller [A(v),B(v),C(v),D(v)]
2D Observer Form [A(v),B(v),C(v),F(v),H(v)]
2D Self-Conditioned [A(v),B(v),C(v),D(v)]
3D Controller [A(v),B(v),C(v),D(v)]
3D Observer Form [A(v),B(v),C(v),F(v),H(v)]
3D Self-Conditioned [A(v),B(v),C(v),D(v)]
Gain Scheduled Lead-Lag
Communications Toolbox - 3
Free Space Path Loss 10 dB
Free Space Path Loss
Cubic Polynomial
Memoryless Nonlinearity
AGC
Control System Toolbox - 11

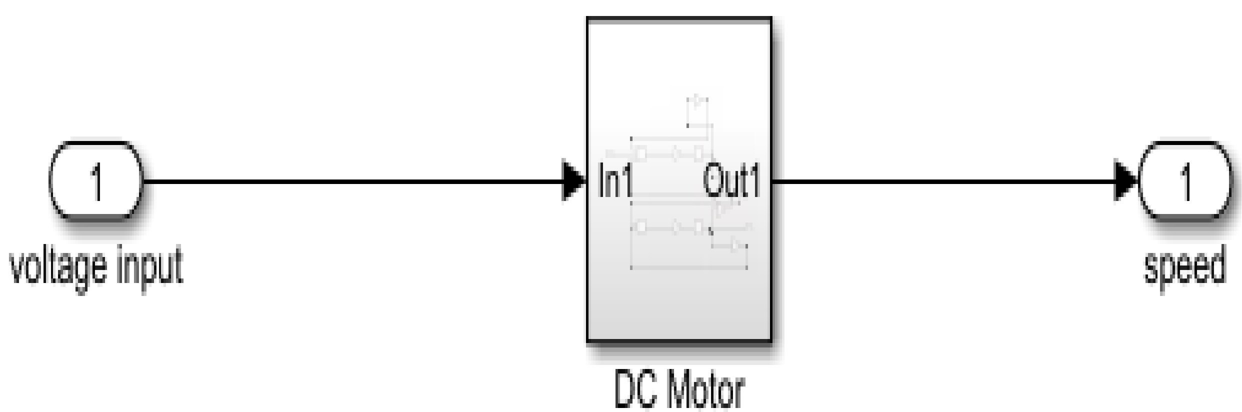
OUTPUT RESPONSE:



Dc motor:




R
1
voltage input
-
+
-
1/L
1
s
i
K
K
+
-
1/J
1
s
1
speed
b


1
voltage input
In1
Out1
DC Motor
1
speed

Matlab code:

J=0.01;

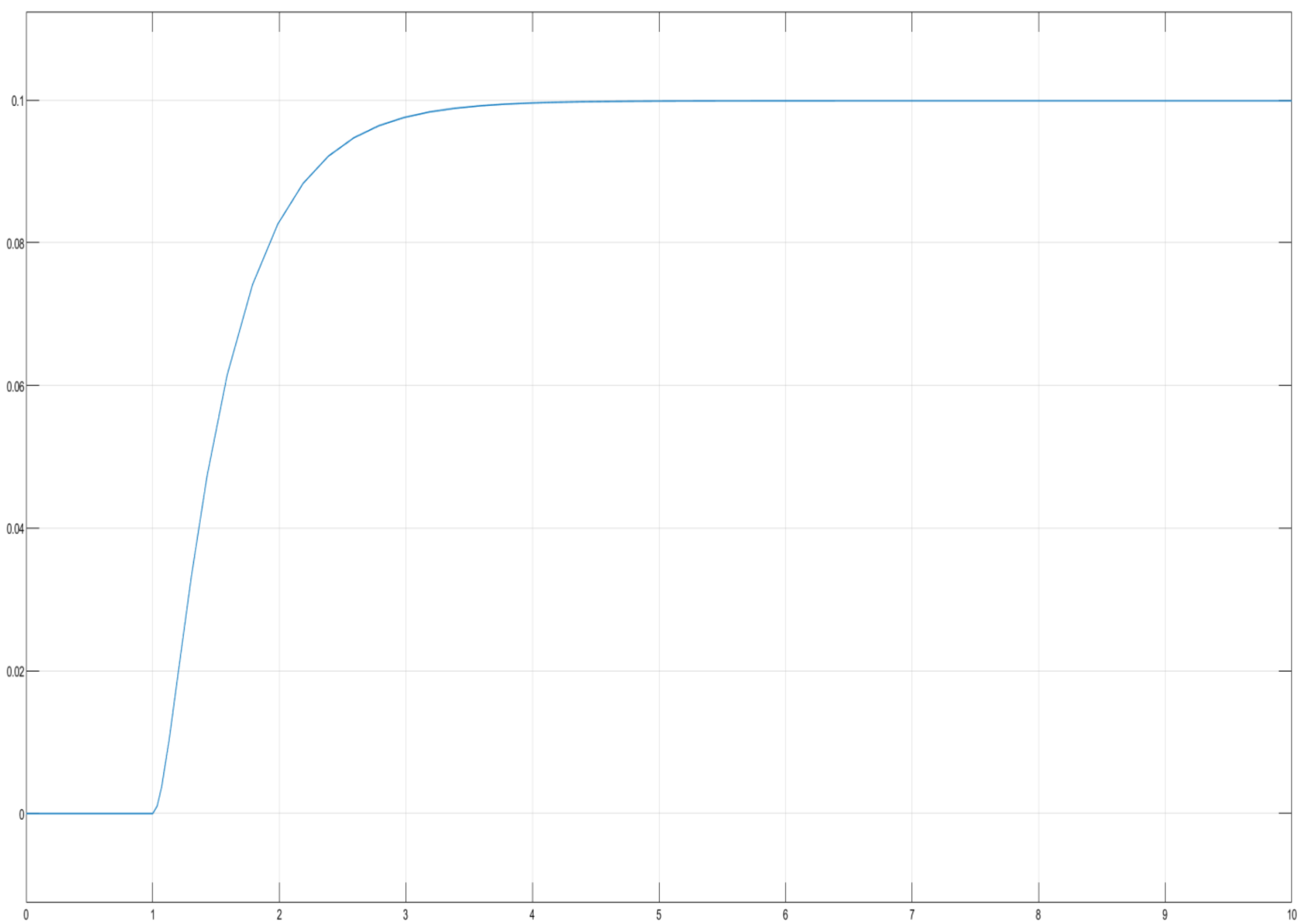
b=0.1;

K=0.01;

R=1;

L=0.5;

Output:



FUTURE SCOPE:

MATLAB simulation for modeling of DC motor has been done which can be implemented in hardware to observe the actual feasibility of the approach applied in this thesis.

This technique can be extended to other types of motors. In this thesis, we have done modeling of DC motor.

CONCLUSION:

The paper presents modeling and simulation of the dynamic equations of electromechanical DC motor in Simulink and simulations are shown graphically in MATLAB Simulink. This validation is done to ensure that modeling can be used to replicate the behavior of a DC motor dynamic.

Results obtained are then compared and found to be in close agreement. The contribution of this work is significantly expedient in the field of automobile, mechatronics, and control systems and educative for hand-on laboratory. It also provides a novel approach to simulating the electromechanical system in a concise and precise method.