



EV MODELLING USING QSS AND ADVISOR TOOLBOX IN MATLAB

**MAYUR MADHUKAR MANKAR, PROF. NARENDRANATH R. BODAKE, PROF.
NILESH G. DAFADE, PROF. AJEET S. YADAV**

Lecturer, Sr. Lecturer, Sr. Lecturer, Lecturer
LoGMIEER Nashik, GGSP NASHIK

INTRODUCTION

Contemporary fuel vehicles are the primary means of transportation for people, providing significant convenience. However, they also test the earth's limited resources and contribute to air pollution through exhaust emissions, adversely affecting public health. Consequently, electric vehicles (EVs) are emerging as a viable alternative. EVs are becoming more popular due to their affordability and lower environmental impact, with companies like Tesla leading the way.

As the electric vehicle industry expands, various data points and performance metrics are being closely monitored. China, in particular, has made significant strides in assisted and unmanned driving technologies. In the Internet era, the development of pure electric vehicles must explore broader opportunities. The synergy between pure electric vehicles, distributed energy, intelligent transportation, and 5G communication technology is paving the way for the advancement of EVs in China, establishing a strong foundation for their future development.

The Simulink simulation model facilitates easy data updates, allowing for timely correction and debugging of electric vehicle characteristics, such as basic mechanisms, performance parameters, and feedback mechanisms. This case's design, modeling, simulation, and testing processes organically integrate to create a comprehensive development mode for automotive embedded systems.

INTRODUCTION TO MATLAB

MATLAB, short for "MATrix LABoratory," is a high-level programming language and interactive environment primarily used for numerical computation, visualization, and programming. Developed by MathWorks, it allows users to perform a wide range of computational tasks, including algorithm

development, data analysis, visualization, and numerical computation. MATLAB is widely used in academia and industry for research, development, and analysis.

Key Features of MATLAB:

High-Level Language:

MATLAB provides a high-level programming language that is easy to learn and use. It includes built-in functions for mathematical operations, data analysis, and graphical visualization.

Interactive Environment:

MATLAB offers an interactive environment that combines a desktop interface tuned for iterative exploration, design, and problem-solving with a programming language that expresses matrix and array mathematics directly.

Mathematical Functions:

MATLAB includes a comprehensive set of mathematical functions for linear algebra, statistics, Fourier analysis, filtering, optimization, numerical integration, and solving ordinary differential equations.

Data Analysis and Visualization:

MATLAB provides tools for acquiring, analyzing, and visualizing data. Users can create 2D and 3D plots, graphs, and charts to visualize data and results.

Toolboxes:

MATLAB has numerous add-on toolboxes for specialized applications such as signal processing, image processing, control systems, neural networks, and more. These toolboxes extend MATLAB's capabilities and are developed by experts in various fields.

Simulink Integration:

MATLAB is integrated with Simulink, a graphical environment for simulation and Model-Based Design of multi-domain dynamic and embedded systems. Simulink provides an interactive environment for modeling, simulating, and analyzing dynamic systems.

Code Generation:

MATLAB supports code generation for creating C, C++, and HDL code from MATLAB algorithms, enabling deployment to embedded systems and integration with other software.

Application Development:

MATLAB provides tools for developing applications with custom graphical user interfaces. It also supports integration with other programming languages and technologies such as Python, Java, .NET, and SQL.

Applications of MATLAB:

Engineering and Scientific Research:

MATLAB is extensively used in various fields of engineering and scientific research for modeling, simulation, and analysis.

Data Science and Machine Learning:

MATLAB provides robust tools for data analysis, machine learning, and deep learning, making it a popular choice for data scientists.

Finance:

MATLAB is used for quantitative analysis, algorithmic trading, risk management, and financial modeling.

Education:

MATLAB is widely used in academic institutions for teaching and research in mathematics, engineering, and science courses.

APPLICATION OF MATLAB IN EV INDUSTRY

MATLAB plays a significant role in the electric vehicle (EV) industry, offering a variety of tools and functionalities to support different aspects of EV development and operation. Here are some key applications of MATLAB in the EV industry:

1.2.1. Battery Management Systems (BMS):

- **Modeling and Simulation:** MATLAB can model and simulate the behavior of battery cells and packs, including state-of-charge (SOC) and state-of-health (SOH) estimation, thermal management, and fault diagnosis.
- **Algorithm Development:** Develop algorithms for battery charging and discharging cycles, cell balancing, and predictive maintenance.

1.2.2. Powertrain Design and Optimization:

- **Electric Motor Control:** Design and optimize control algorithms for electric motors, including torque control, speed control, and efficiency optimization.
- **Power Electronics:** Simulate and analyze power electronics components such as inverters, converters, and controllers used in EV powertrains.

1.2.3. Energy Management Systems:

- **Energy Efficiency:** Optimize energy consumption for different driving conditions and develop strategies for regenerative braking.
- **Hybrid Systems:** Model and simulate hybrid electric vehicle (HEV) systems to balance the power distribution between the internal combustion engine and electric motor.

1.2.4. Vehicle Dynamics and Control:

- **Vehicle Simulation:** Simulate vehicle dynamics to analyze performance, stability, and handling characteristics.
- **Control Systems:** Design control systems for stability control, traction control, and advanced driver-assistance systems (ADAS).

1.2.5. Charging Infrastructure:

- **Wireless Charging:** Develop and simulate wireless charging systems, including the design of coils and the optimization of power transfer efficiency.
- **Smart Charging:** Implement smart charging algorithms that can interact with the grid, optimize charging times, and reduce costs.

1.2.6. Autonomous Driving and ADAS:

- **Perception Systems:** Develop algorithms for sensor fusion, object detection, and tracking using data from LiDAR, radar, and cameras.

- **Path Planning and Control:** Design path planning algorithms and vehicle control systems for autonomous driving.

1.2.7. Thermal Management:

- **Cooling Systems:** Model and simulate thermal management systems to ensure optimal temperature control of batteries, motors, and power electronics.
- **Heat Transfer Analysis:** Perform heat transfer analysis to design efficient cooling strategies for various EV components.

1.2.8. Data Analysis and Machine Learning:

- **Predictive Maintenance:** Use machine learning algorithms to predict component failures and schedule maintenance.
- **Performance Monitoring:** Analyze vehicle performance data to identify trends, optimize performance, and enhance reliability.

1.2.9. Prototyping and Testing:

- **Hardware-in-the-Loop (HIL) Simulation:** Integrate MATLAB with HIL systems to test and validate control algorithms and hardware components in real-time.
- **Rapid Prototyping:** Quickly prototype and iterate on designs using MATLAB and Simulink to shorten development cycles.

1.2.10. Compliance and Standards:

- **Regulatory Compliance:** Ensure that EV components and systems meet industry standards and regulatory requirements through modeling and simulation.
- **Safety Analysis:** Perform safety analysis and verification of control systems to comply with automotive safety standards like ISO 26262.

1.2.11. System Integration:

- **Communication Protocols:** Develop and test communication protocols such as CAN, LIN, and FlexRay for seamless integration of EV components.
- **Multi-domain Modeling:** Integrate mechanical, electrical, and software components into a unified simulation environment to study interactions and optimize overall system performance.

In summary, MATLAB provides a comprehensive set of tools that are essential for various stages of EV development, from conceptual design and simulation to testing and deployment. This versatility makes MATLAB an indispensable tool in the EV industry, supporting innovation and enhancing efficiency in electric vehicle development.

EV MODELING IN MATLAB USING QSS TOOLBOX

For modeling in matlab we need to specify some parameters of vehicle which is we are going to design.

The parameters are as following.

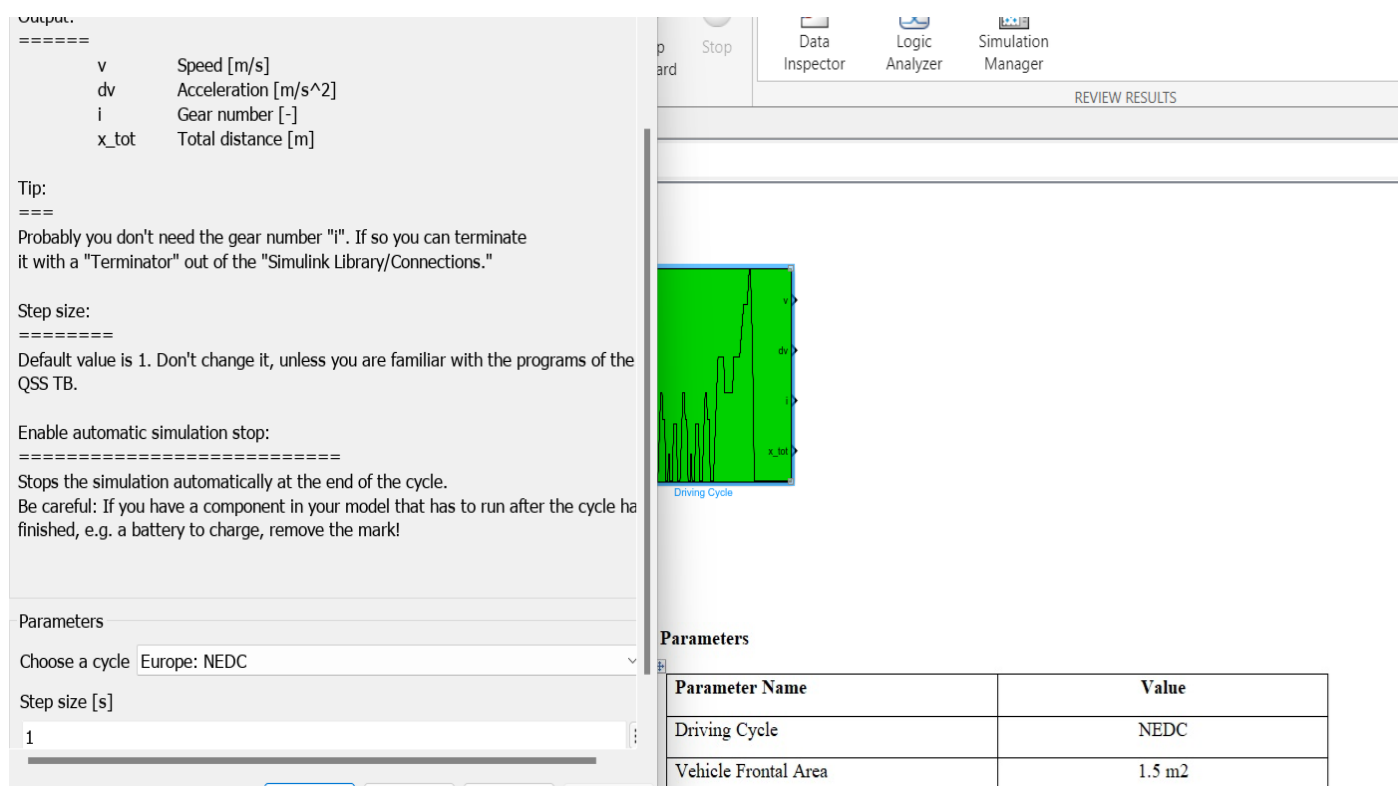
Vehicle Parameters

Table No.2.1.Vehicle Parameters

Parameter Name	Value
Driving Cycle	NEDC
Vehicle Frontal Area	1.5 m ²
Air Drag Co-efficient	0.5
Rolling Resistance Co-efficient	0.015
Wheel Diameter	0.3m
Gear Ratio	2
Transmission Efficiency	98%
Motor Map	'qss_em_origional_map'
Motor Inertia	0.1 kg.m ²
Cells in Series	13
Cells in Parallel	1

Insertion of Driving cycle in Simulink

Here is insertion of drive cycle and specified it as NEDC driving cycle.



Output:

```

=====
v      Speed [m/s]
dv     Acceleration [m/s^2]
i      Gear number [-]
x_tot  Total distance [m]
  
```

Tip:
====
Probably you don't need the gear number "i". If so you can terminate it with a "Terminator" out of the "Simulink Library/Connections."

Step size:
=====
Default value is 1. Don't change it, unless you are familiar with the programs of the QSS TB.

Enable automatic simulation stop:
=====
Stops the simulation automatically at the end of the cycle.
Be careful: If you have a component in your model that has to run after the cycle has finished, e.g. a battery to charge, remove the mark!

Parameters

Choose a cycle: Europe: NEDC

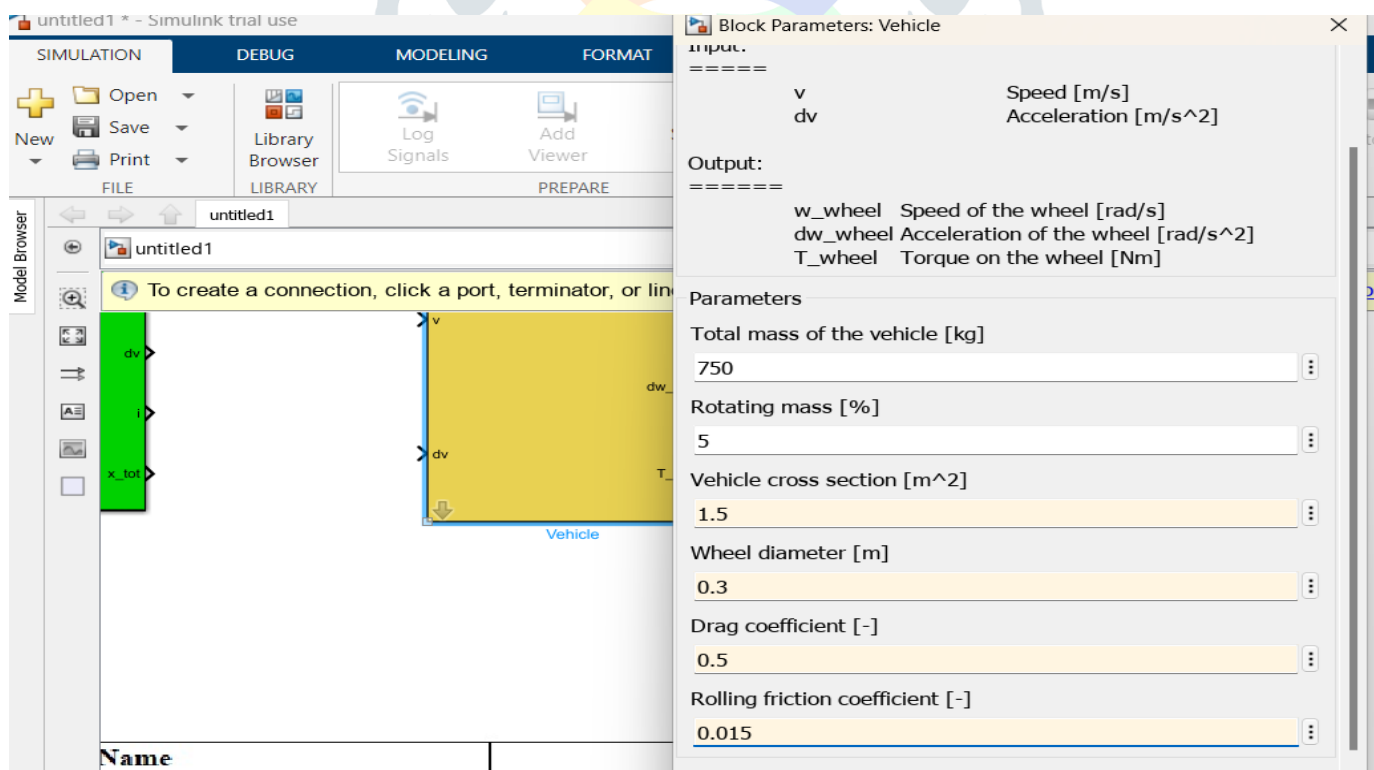
Step size [s]: 1

Parameter Name	Value
Driving Cycle	NEDC
Vehicle Frontal Area	1.5 m2

Fig.2.1.Driving cycle insertion

Insertion of vehicle block in simulink

Here is insertion of Vehicle Block and specified it as per mentioned table.



untitled1 * - Simulink trial use

SIMULATION | DEBUG | MODELING | FORMAT

Open | Save | Print | FILE | Library Browser | LIBRARY | Log Signals | Add Viewer | PREPARE

Model Browser: untitled1

To create a connection, click a port, terminator, or line

Block Parameters: Vehicle

Input:

```

=====
v      Speed [m/s]
dv     Acceleration [m/s^2]
  
```

Output:

```

=====
w_wheel Speed of the wheel [rad/s]
dw_wheel Acceleration of the wheel [rad/s^2]
T_wheel Torque on the wheel [Nm]
  
```

Parameters

Total mass of the vehicle [kg]: 750

Rotating mass [%]: 5

Vehicle cross section [m^2]: 1.5

Wheel diameter [m]: 0.3

Drag coefficient [-]: 0.5

Rolling friction coefficient [-]: 0.015

Fig.2.2. Vehicle Block

Insertion of Simple Transmission Block

er

Table

Fast Restart

Back

SIMULATE

ator, or line segment, and then click a compatible, highlighted mod

w_wheel

dw_wheel

T_wheel

w_wheel

dw_wheel

T_wheel

Simple Transmission

Value
NEDC

Block Parameters: Simple Transmission

Simple Transmission (mask) (link)

This block simulates a transmission with a fixed gear ratio.

Input:
=====

w_wheel

dw_wheel

T_wheel

Speed of the wheel [rad/s]

Acceleration of the wheel [rad/s^2]

Torque on the wheel [Nm]

Output:
=====

w_trans

dw_trans

T_trans

Speed of the fly wheel [rad/s]

Acceleration of the fly wheel [rad/s^2]

Torque on the fly wheel [Nm]

Parameters

Gear ratio [-]

2

Efficiency [-]

0.98

Idling losses (friction) [W]

50

Minimum wheel speed beyond which losses are generated [rad/s]

1

OK

Cancel

Help

Apply

Fig.2.3.Transmission Block

Insertion of Electric Motor

hted model element. [More information.](#) [Do not sho](#)

w_trans

dw_trans

T_trans

w_gear

dw_gear

T_gear

P_EM

Electric Motor

This block simulates the behaviour of an electric motor. The block is based on an efficiency map.

Input:
=====

w_gear

dw_gear

T_gear

Speed of the fly wheel [rad/s]

Acceleration of the fly wheel [rad/s^2]

Torque on the fly wheel [Nm]

Output:
=====

P_EM

Power produced by the electric motor [W]

Parameters

Motor scaling factor [-]

1

Motor inertia [kg*m2]

0.1

Power required by auxiliaries [W]

0

OK

Cancel

Help

Apply

Fig.2.4.Electric motor Block

Insertion of Battery

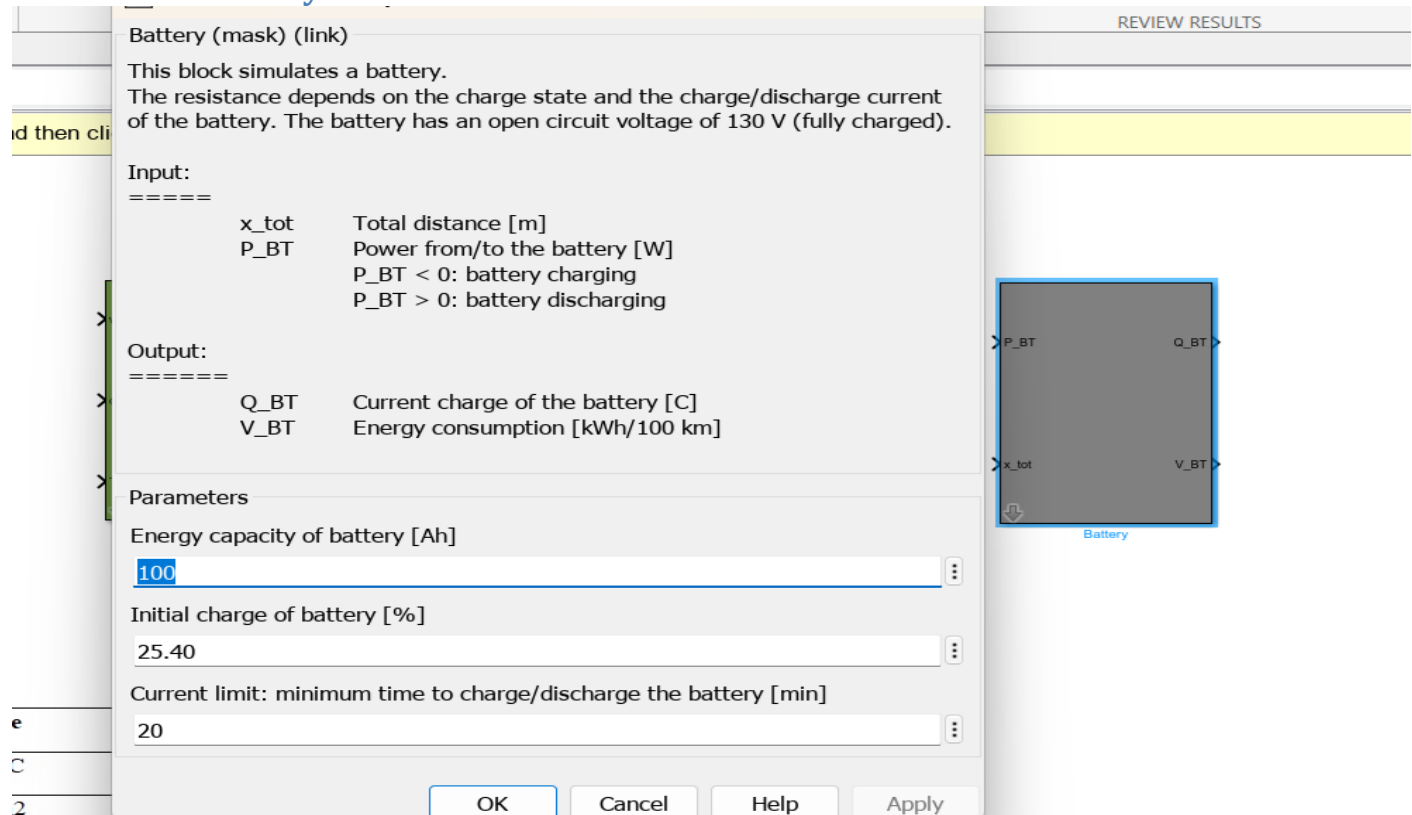


Fig.2.5. Battery Block

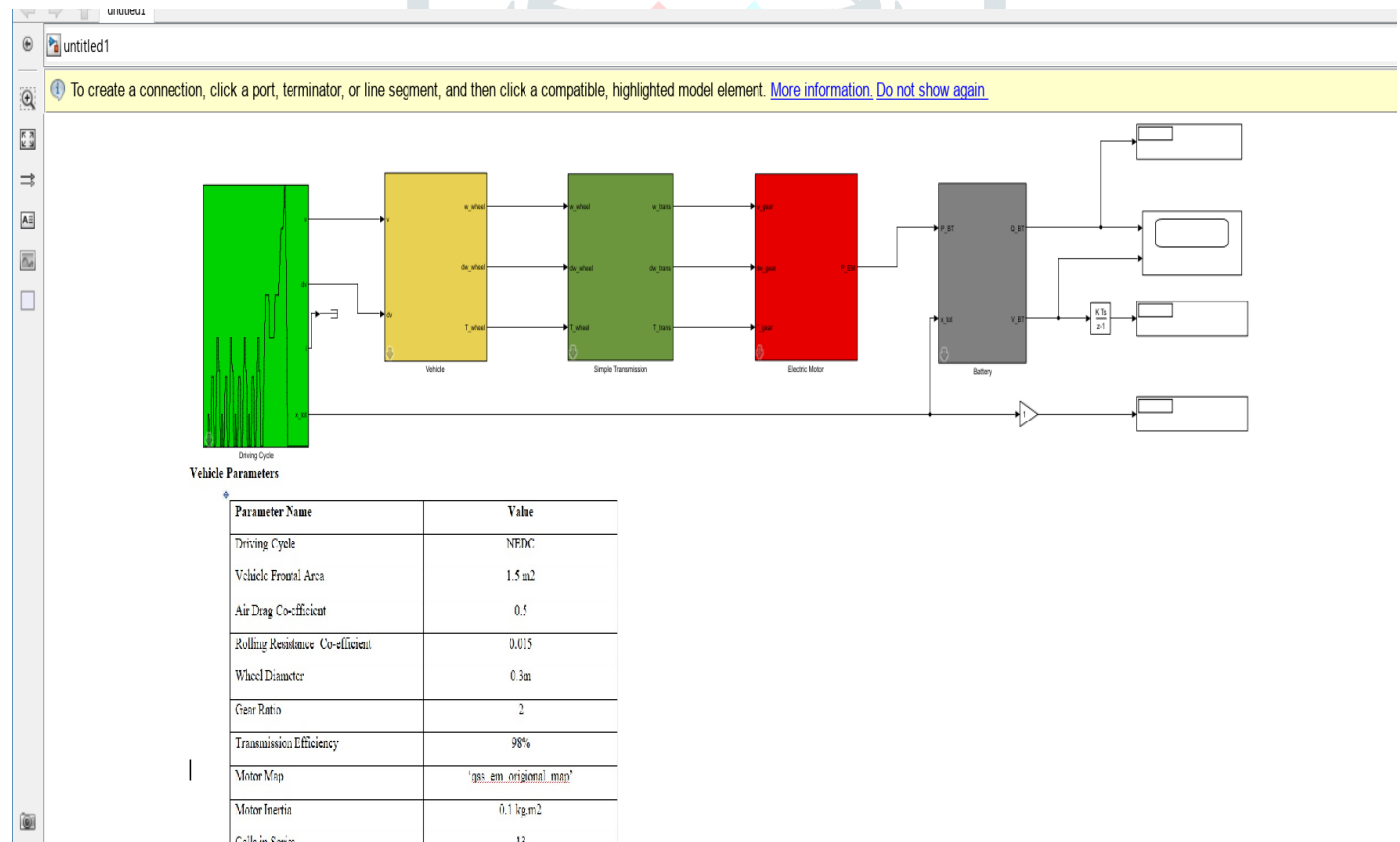
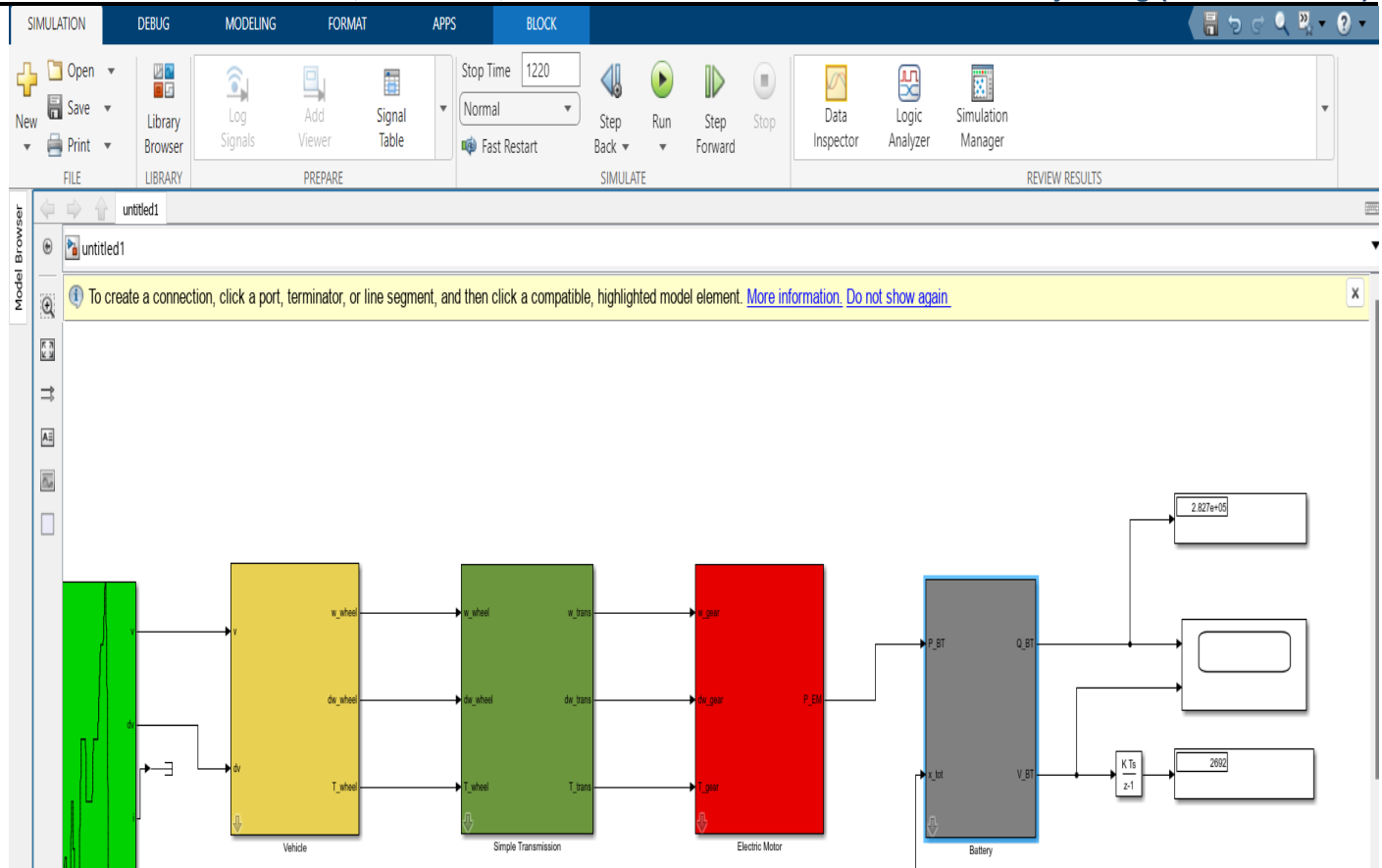
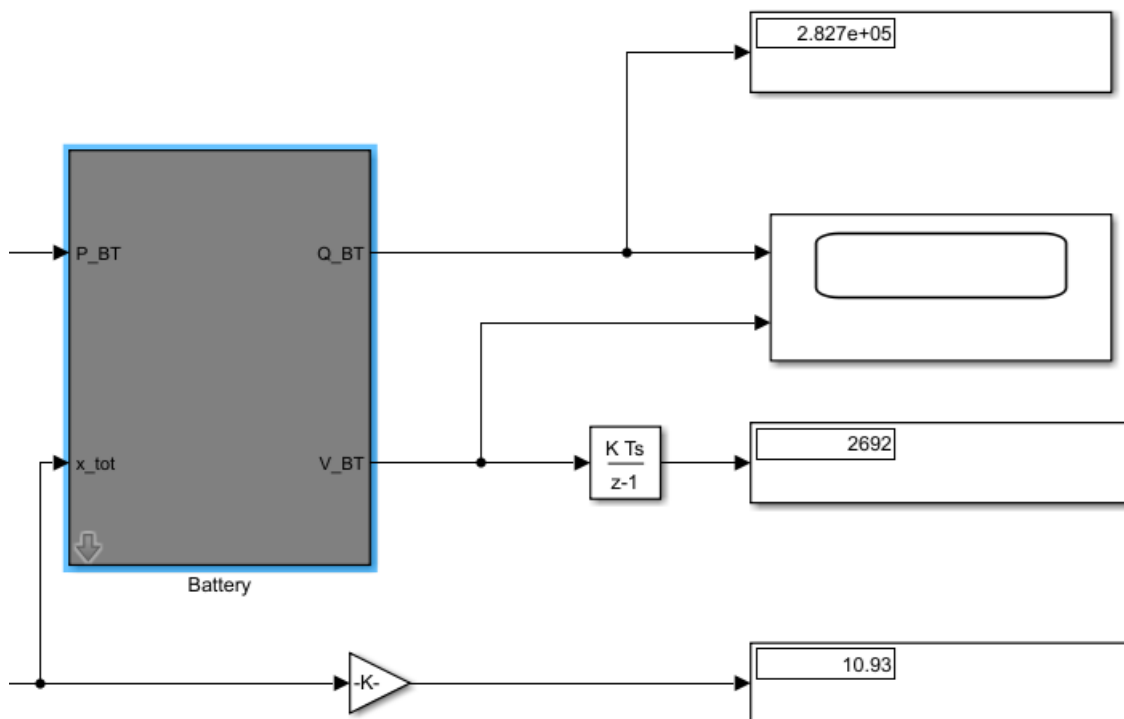
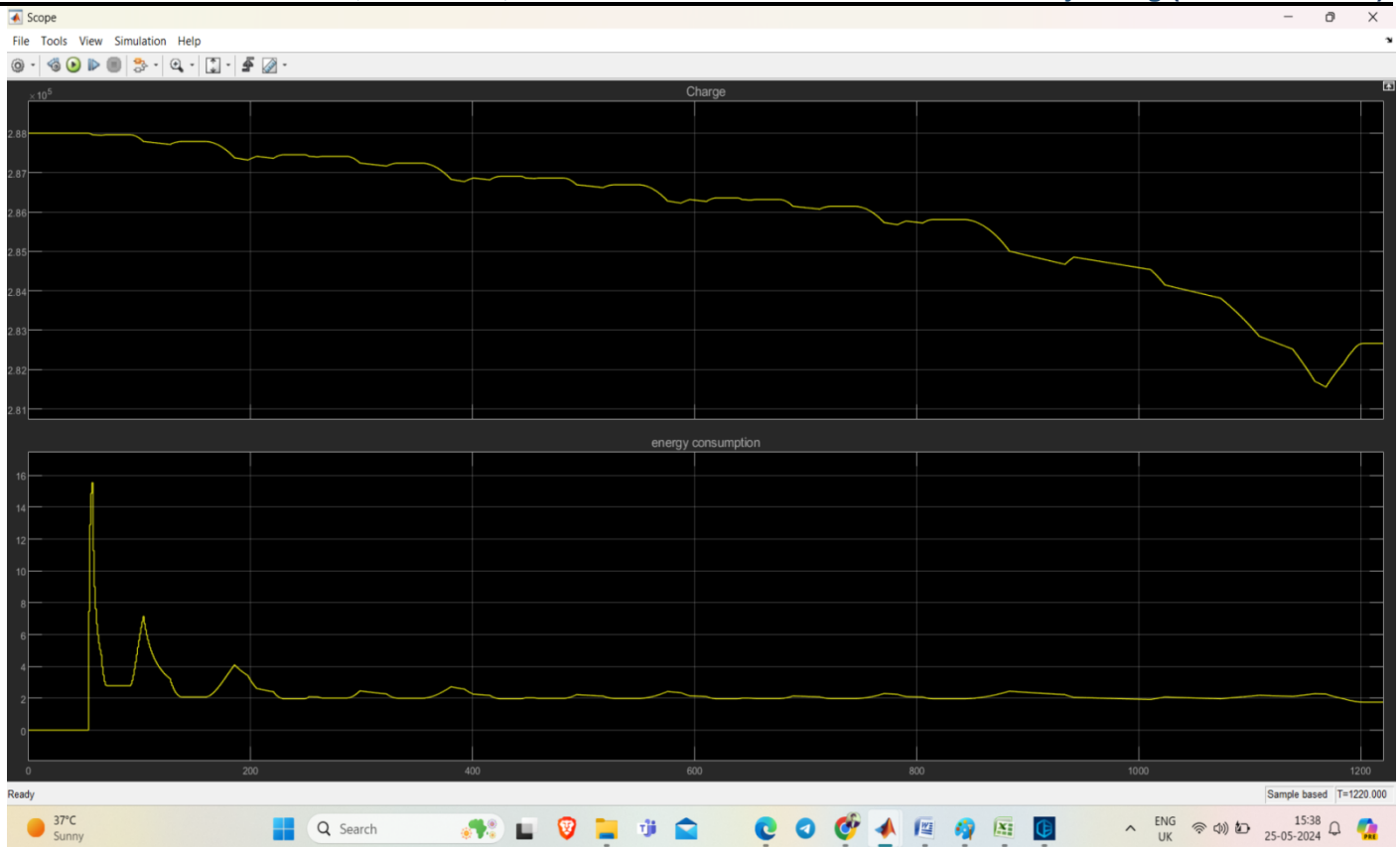
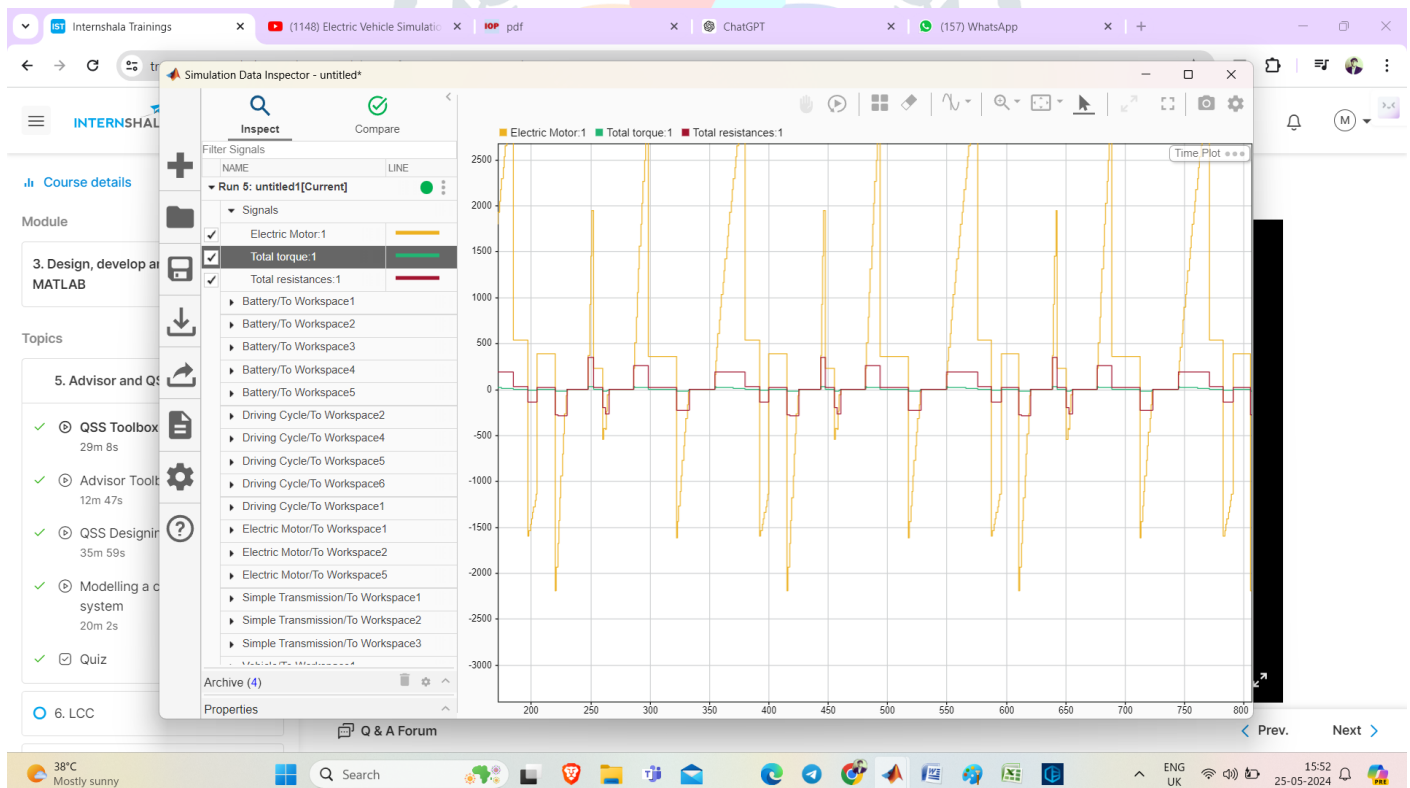


Fig.2.6. Final Block model in Simulink

Fig.2.7. **Results** Obtained After running the simulationFig.2.8. Simulink **results** after running simulation

Fig.2.9. Scope **results** after running the simulationFig.2.10. **Result** obtained in Data Inspector

After the completion of this simulation we did same simulation by changing the input parameters. These results are tabulated as follows.

Simulation. 1**Input Parameters****EV MODELING IN MATLAB USING ADVISOR TOOLBOX**

Fig. 3.1. Advisor toolbox Selection of EV model

Parameter Name	Value
Driving Cycle	NEDC
Vehicle Frontal Area	1.5 m ²
Air Drag Co-efficient	0.5
Rolling Resistance Co-efficient	0.015
Wheel Diameter	0.3m
Gear Ratio	2
Transmission Efficiency	98%
Motor Map	'qss_em_origional_map'
Motor Inertia	0.1 kg.m ²
Cells in Series	13
Cells in Parallel	1

Load File EV_defaults_in

Drivetrain Config ev

	version	?	type	
Vehicle		?		VEH_SMCAR
Fuel Converter		?		VEH_PRIUS_JPN
Exhaust Aftertreat		?		VEH_RTS06
Energy Storage	rint	?	pb	VEH_SMCAR
Energy Storage 2		?		VEH_SUBURBAN_RWD
Motor		?		VEH_SUV
Motor 2		?		VEH_SUV_RWD
Starter		?		VEH_TT
Generator		?		VEH_cargoVan
Transmission	man	?	man	VEH_compact
Transmission 2		?		VEH_largeCar
Clutch/Torq. Conv.		?		VEH_largeSUV
Torque Coupling		?		VEH_midSizeCar
Wheel/Axle	Crr	?	Crr	VEH_midSizeSUV
Accessory	Const	?	Const	VEH_minivan
Acc Electrical		?		VEH_ralphs_grocery
Powertrain Control	ev	?	man	VEH_smallSUV

max pwr (kW)

#of mod

25

75

Cargo M

Fig.3.2. Selection of type of vehicle

Vehicle		?		VEH_SMCAR	592		
Fuel Converter		?		fc options			
Exhaust Aftertreat		?		EX_CI			
Energy Storage	rint	?	pb	ESS_PB25	25	308	275
Energy Storage 2		?		ESS_NULL			
Motor		?		ESS_PB104	75	0.92	91
Motor 2		?		ESS_PB12			
Starter		?		ESS_PB16			
Generator		?		ESS_PB18			
Transmission	man	?	man	ESS_PB25			
Transmission 2		?		ESS_PB25_IDEAL			
Clutch/Torq. Conv.		?		ESS_PB28			

#of mod

V nom

1

50

Fig.3.3. Selection of type of Motor in Advisor toolbox

<input type="checkbox"/>	Energy Storage 2		?		ess 2 options	
<input checked="" type="checkbox"/>	Motor		?		MC_AC75	75
	Motor 2		?		motor 2 options	
	Starter		?		starter options	
<input type="checkbox"/>	Generator		?		gc options	
<input checked="" type="checkbox"/>	Transmission	man	?	man	TX_1SPD	
	Transmission 2		?		TX_1SPD	
	Clutch/Torq. Conv.		?		TX_1SPD_BUS	
<input type="checkbox"/>	Torque Coupling		?		TX_1SPD_Focus	
<input checked="" type="checkbox"/>	Wheel/Axle	Crr	?	Crr	TX_1SPD_IDEAL	
<input checked="" type="checkbox"/>	Accessory	Const	?	Const	TX_5SPD	
	Acc Electrical		?		TX_5SPD_CI	
<input checked="" type="checkbox"/>	Powertrain Control	ev	?	man	TX_5SPD_IDEAL	
<input checked="" type="radio"/> front wheel drive <input type="radio"/> rear wheel drive						
View Block Diagram						BD_EV
						TX_5SPD_SI TX_5SPD_SI_INSIGHT TX_AnnexVII_SerHyb TX_AnnexVII_conv TX_MT643_MT TX_RM10145A TX_RT11710B TX_RTLO12610B TX_RTX13710C TX_ZF4HP590

Fig.3.4. Selection of type of Gear box

	Transmission 2		?		trans 2 options
	Clutch/Torq. Conv.		?		clutch/torque converter options
<input type="checkbox"/>	Torque Coupling		?		TC_DUMMY
<input checked="" type="checkbox"/>	Wheel/Axle	Crr	?	Crr	WH_SMCAR
<input checked="" type="checkbox"/>	Accessory	Const	?	Const	WH_CYCLE
	Acc Electrical		?		WH_FOCUS_REGEN
<input checked="" type="checkbox"/>	Powertrain Control	ev	?	man	WH_HEAVY
<input checked="" type="radio"/> front wheel drive <input type="radio"/> rear wheel drive					
View Block Diagram					
BD_EV					
WH_INSIGHT WH_PNGV WH_PRIUS_JPN WH_SMCAR WH_SMCAR_IDEAL WH_SMCAR_REGEN WH_SUV WH_TractorTrailer_highRR WH_TractorTrailer_lowRR WH_TractorTrailer_lowRR_SS WH_TractorTrailer_midRR					
Variable List:					
Component	motor_controller				Edit Var.

Fig.3.5. Selection of type of axle

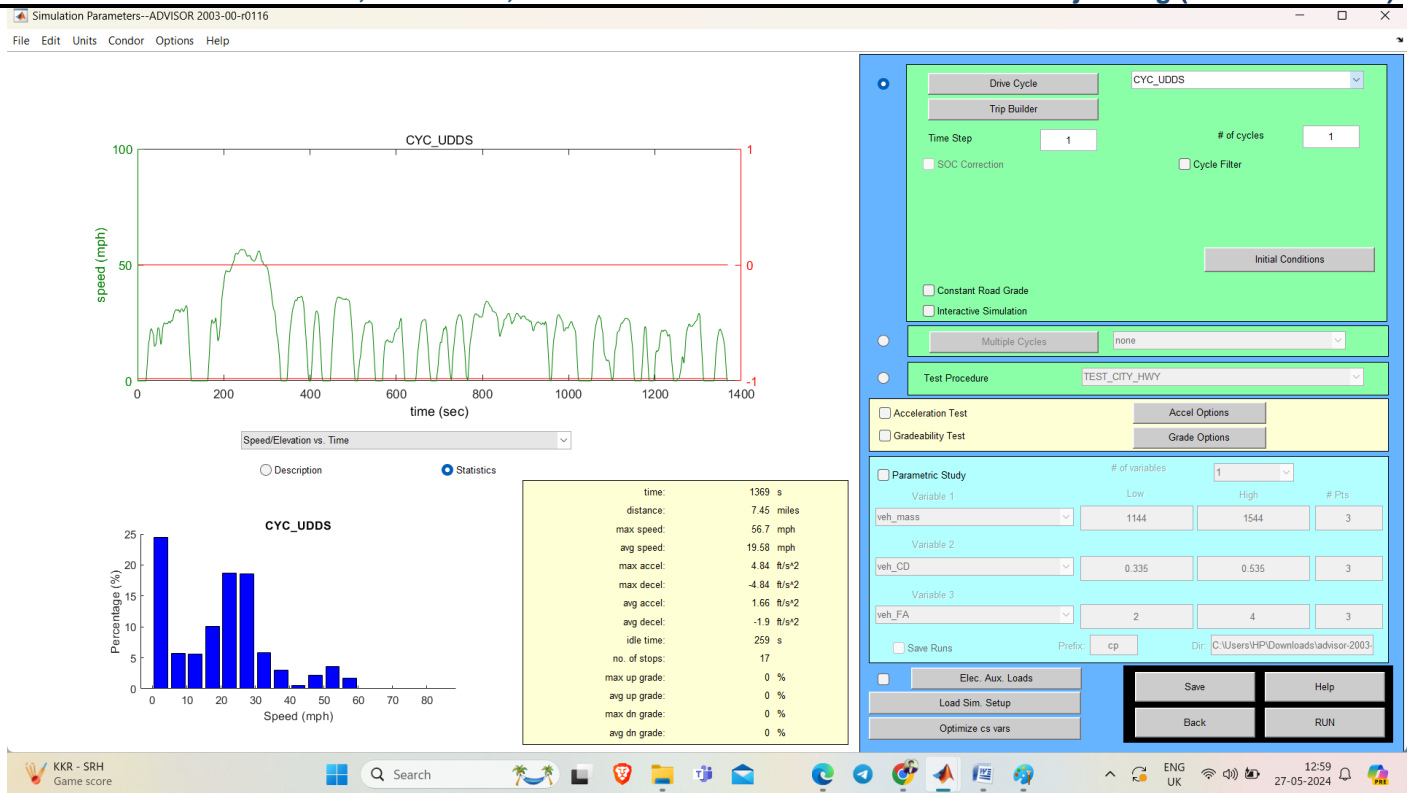


Fig.3.6. EV modelling using Advisor Toolbox

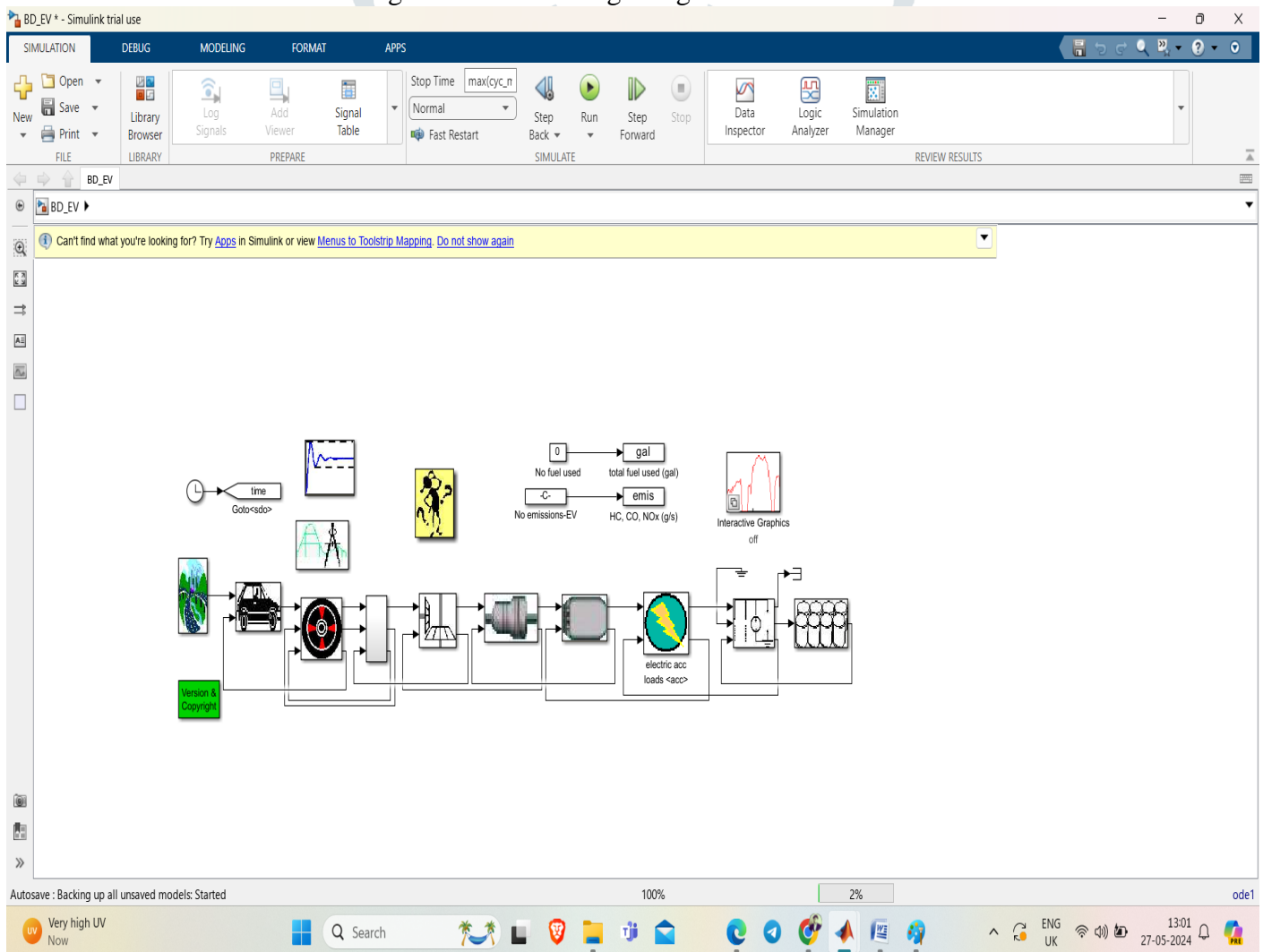


Fig.3.7. EV block Diagram

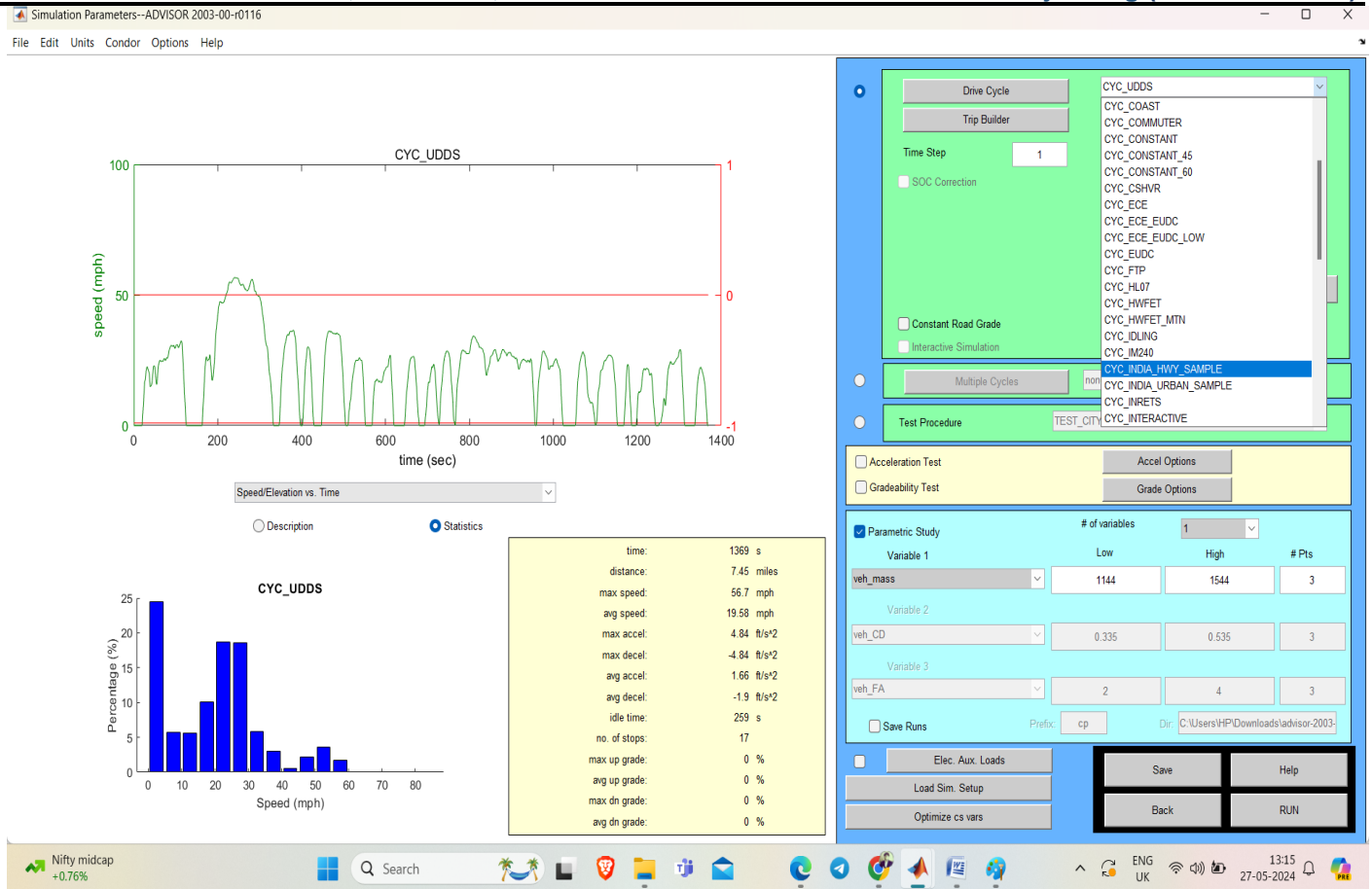


Fig.3.8. Selection of driving condition India Highway Sample

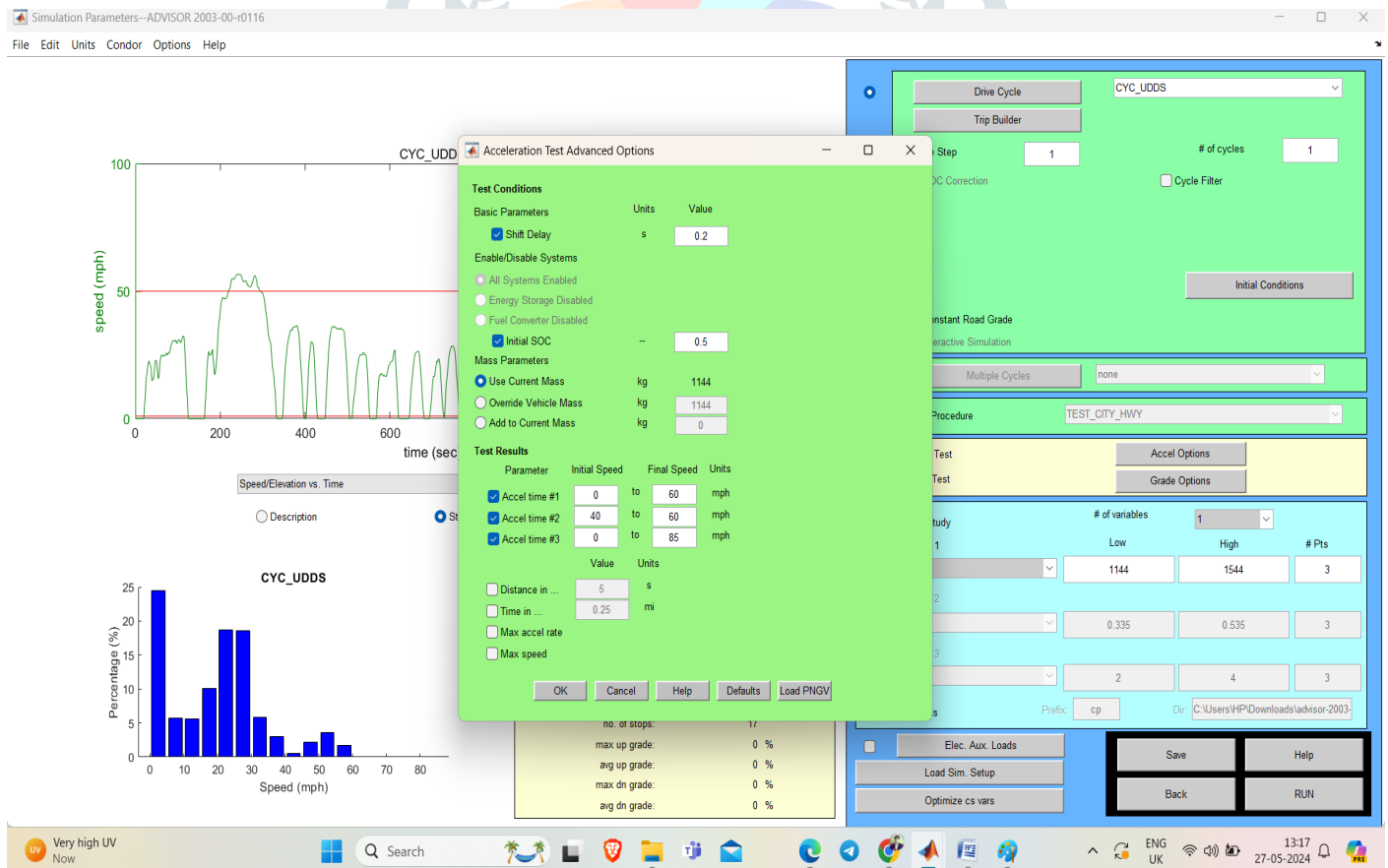


Fig.3.9. Checking Accel Options

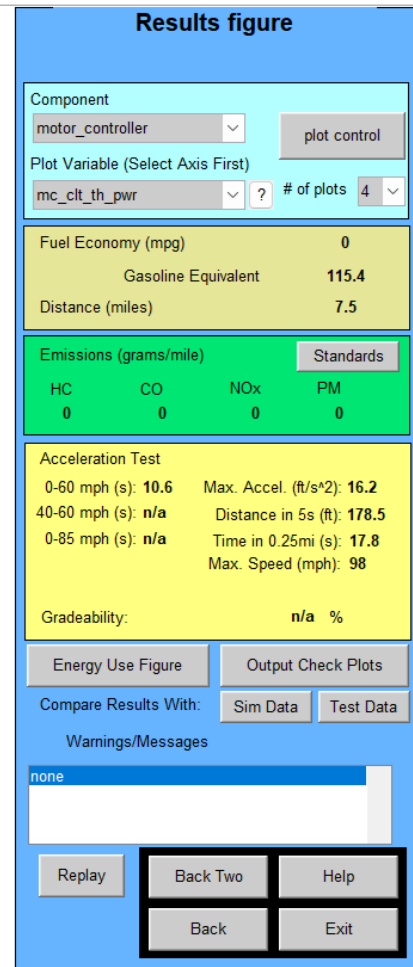
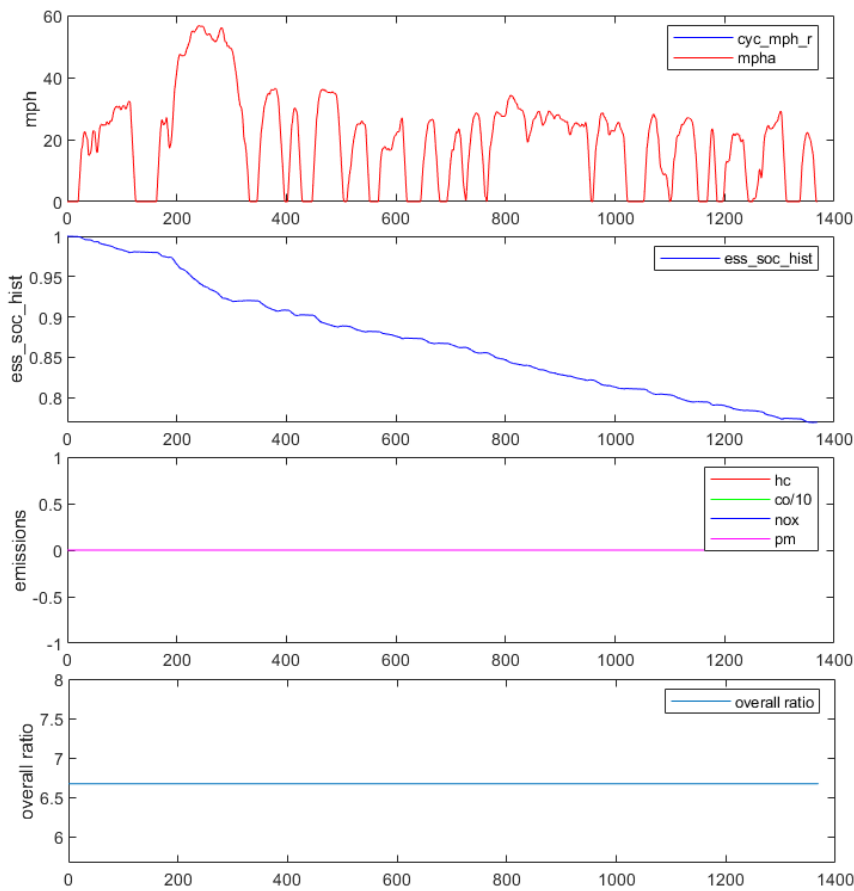
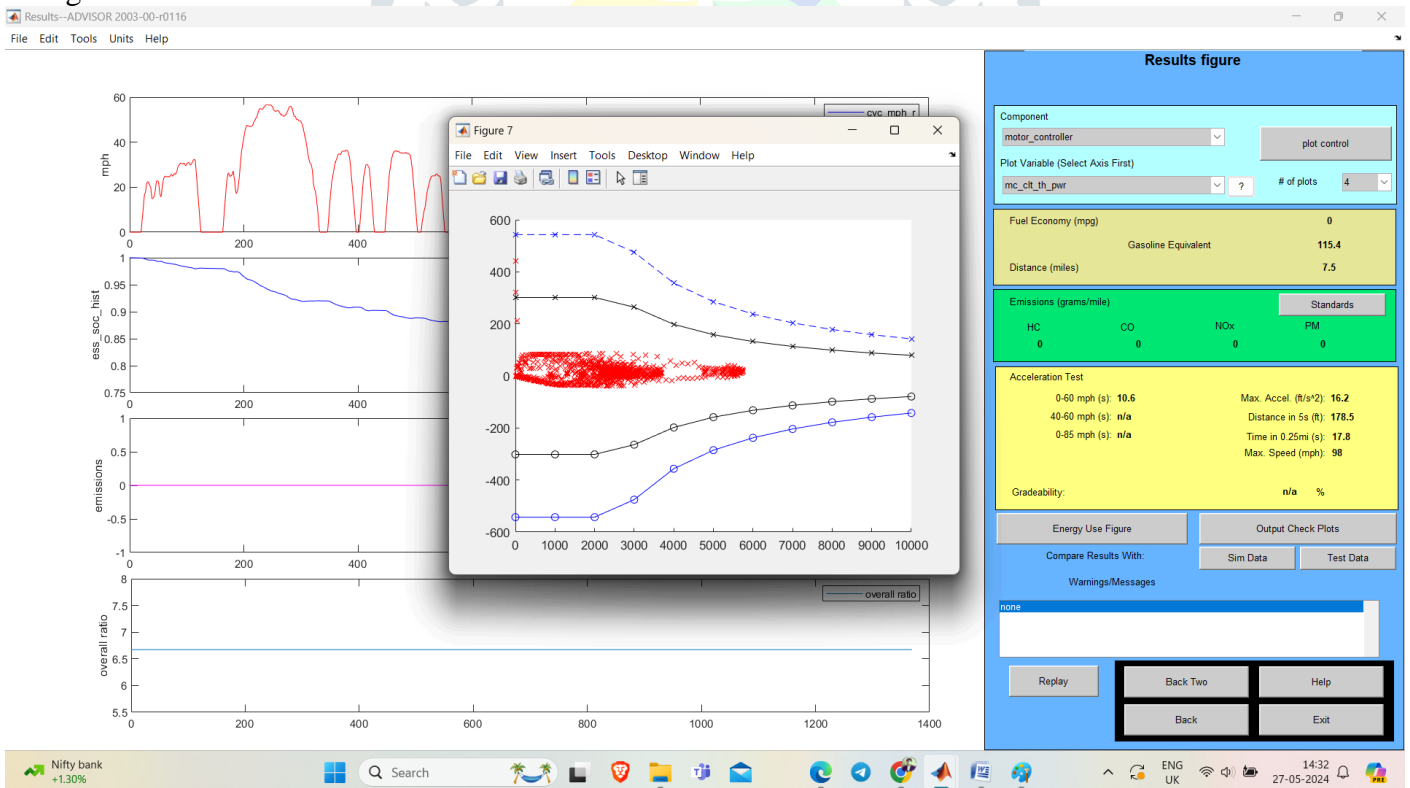
Fig.3.10. **Results** after simulation in Advisor toolbox

Fig.3.11.Plot

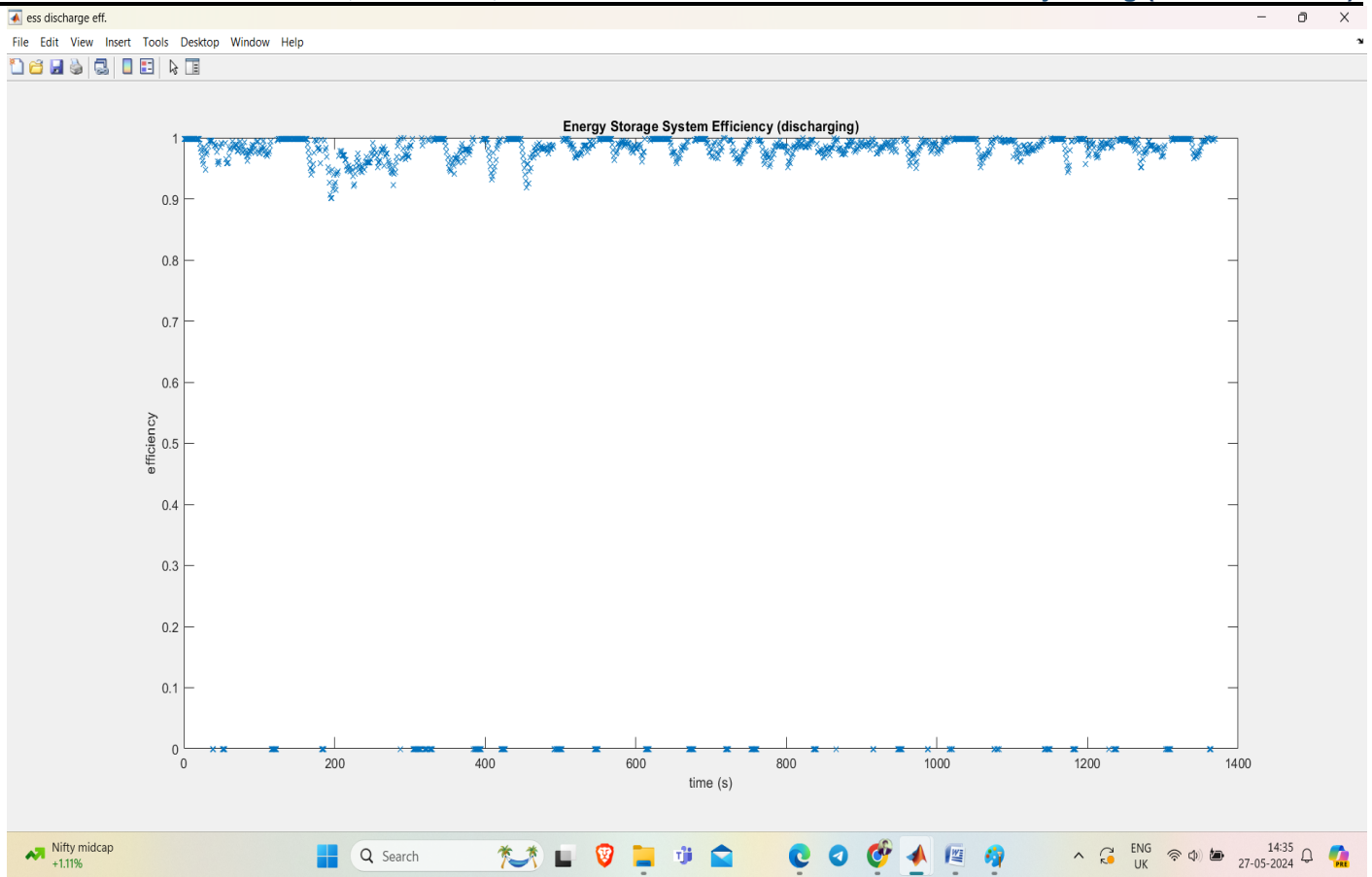


Fig.3.12. time vs efficiency **results** plot Energy storage efficiency (Charging system)

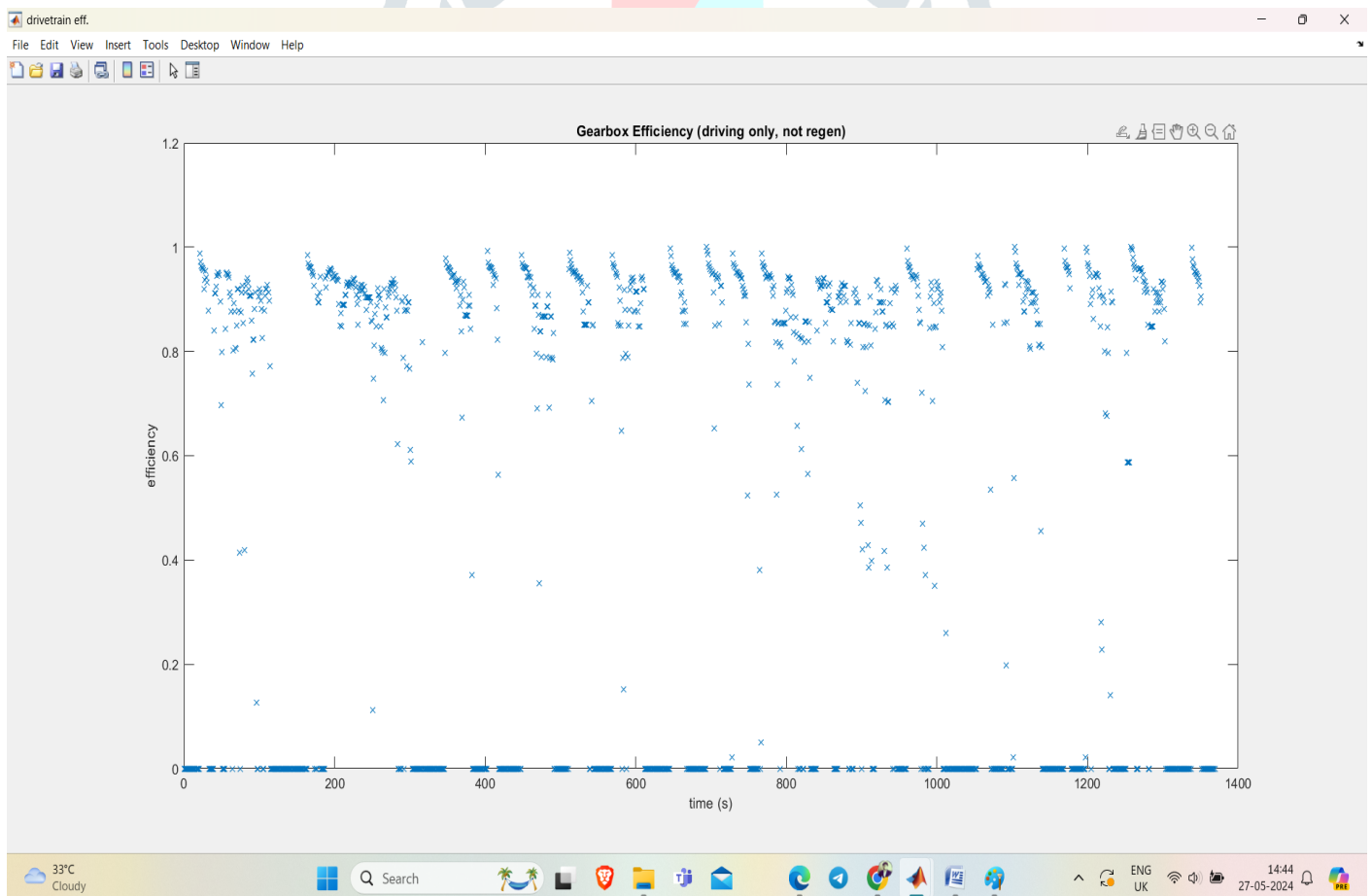


Fig.3.13. **Result** plot Gearbox Efficiency (efficiency vs. time)

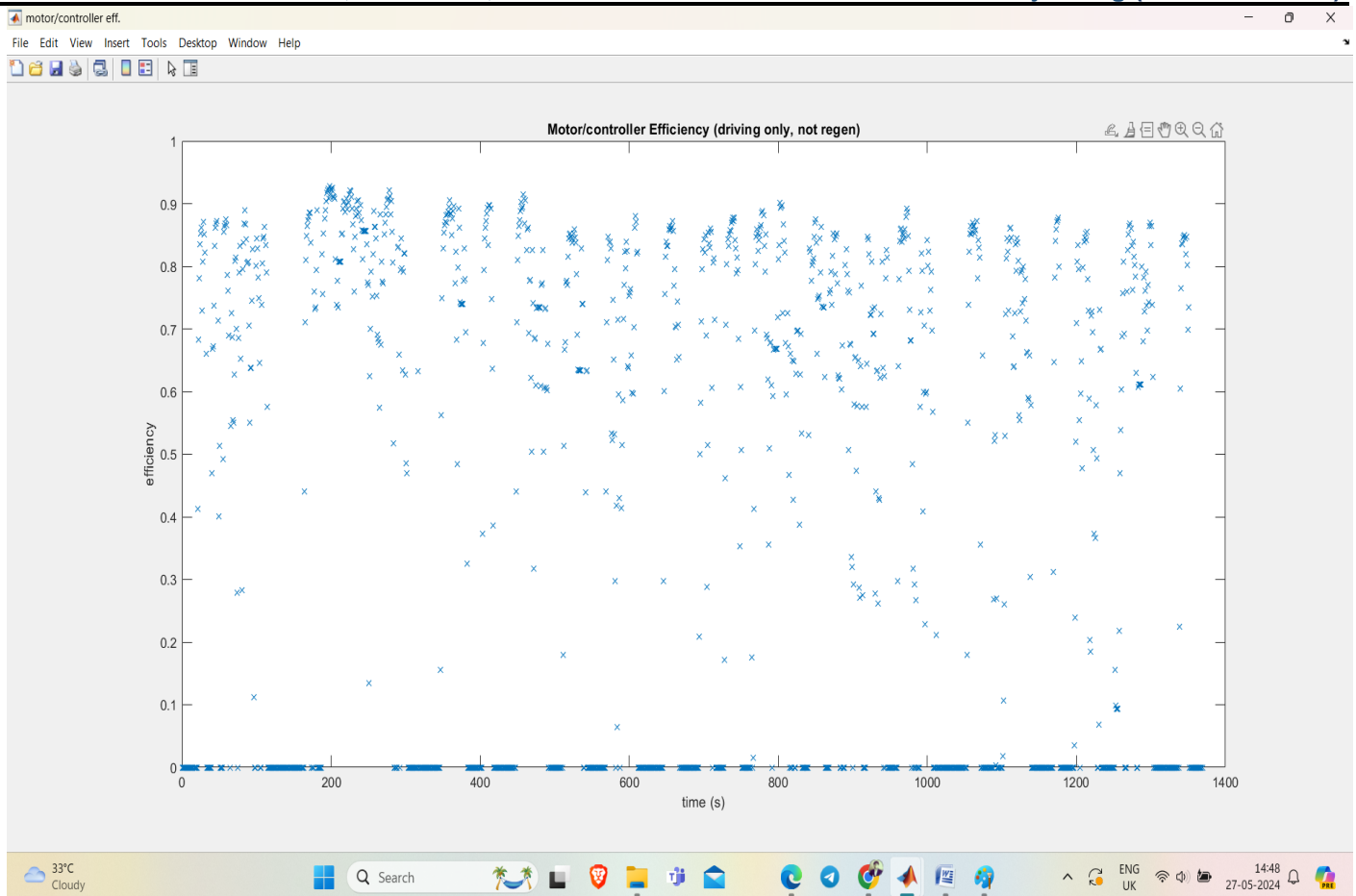


Fig.3.14. Result plot Motor controller Efficiency (efficiency vs. time)

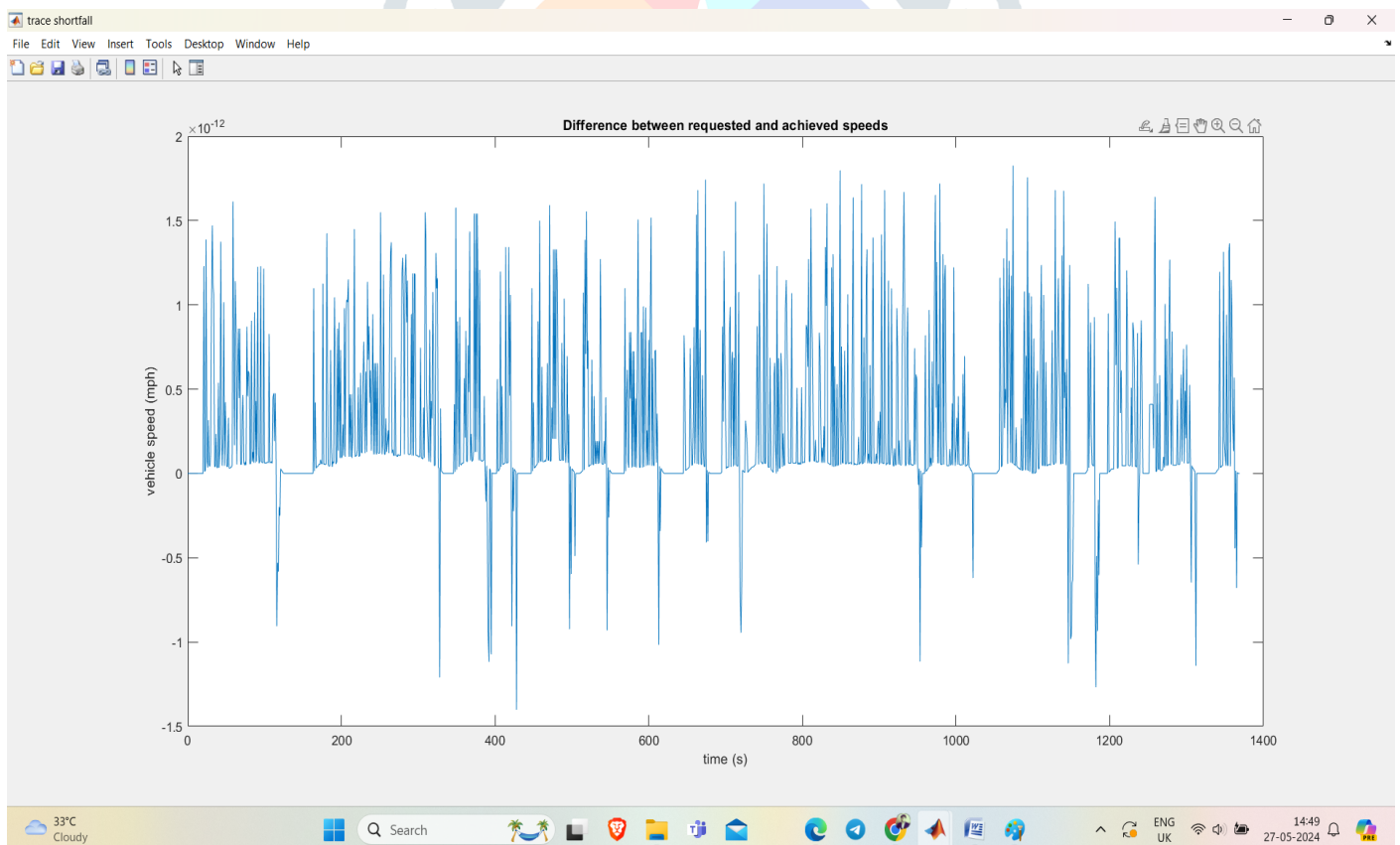


Fig.3.15. Difference between requested and achieved speed

OVERVIEW OF PROJECT

Modeling electric vehicles (EVs) in MATLAB using the QSS (Quasi-Steady-State) and ADVISOR (Advanced Vehicle Simulator) toolboxes involves simulating various aspects of EV performance, energy consumption, and efficiency. Here's an overview of what you can expect from using these toolboxes:

QSS Toolbox

The QSS Toolbox in MATLAB is designed for simulating and analyzing quasi-steady-state behavior of power systems and power electronics. When applied to EV modeling, it provides the following benefits:

1. **Simplified Dynamic Simulation:** QSS focuses on steady-state behavior, which simplifies the simulation of complex systems by avoiding detailed time-domain dynamics.
2. **Efficiency:** Faster simulation times compared to full dynamic models, making it suitable for preliminary design and analysis.
3. **Power Flow Analysis:** Effective for analyzing power flow in EV systems, including battery management, power electronics, and motor performance.
4. **Component Modeling:** Allows for the modeling of individual components like batteries, inverters, and motors in a quasi-steady-state framework.

