CHAPTER 6

EXPERIMENTATION

The experiment is performed in order to validate the modelling and simulation results. In the experimentation process the developed small scale thermionic regenerator is tested. A test setup is developed for experimentation of thermionic conversion. During the experiments different emitter materials are selected and their performance at specified temperatures is observed.

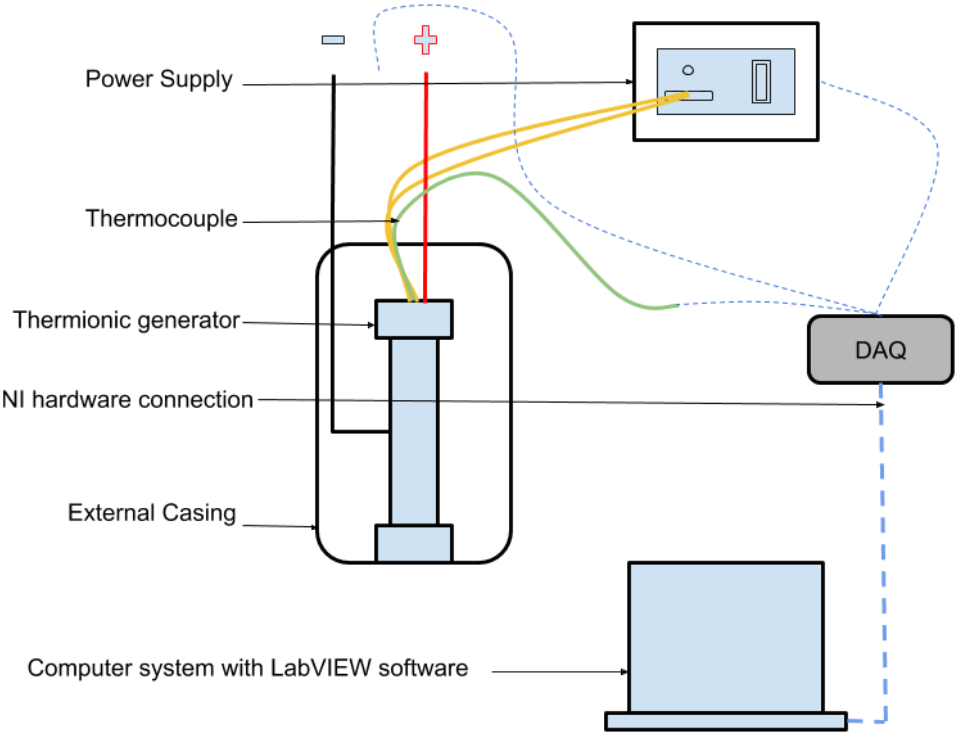


Fig. 6.1. Test Setup Block Diagram

The block diagram shown in Fig. 6.1 is the test setup connections for conducting experimental analysis.

1. Thermionic conversion system: this includes a thermionic generator having assembly of fixtures, guides, emitter and collector with arrangements of heating, vacuuming and electrical connections.
2. Load: for output current measurement a variable resistor is to be used to apply load.
3. Power supply: thermionic generator is to be heated using electric heater using external electric supply. The type of heater used in this setup is cartridge heater.

4. Data acquisition (DAQ): It's the process of taking samples of signals that measure real-world physical circumstances and transforming them into digital numeric values that can be modified by a computer. The emitter temperature, V-I characteristics, and energy supplied are all measured in real time through the hardware connection.
5. Software: national instruments LabVIEW software is proposed for simulations. The program designed on LabVIEW can be connected to this test setup using hardware (DAQ) and analyse it on computer system.

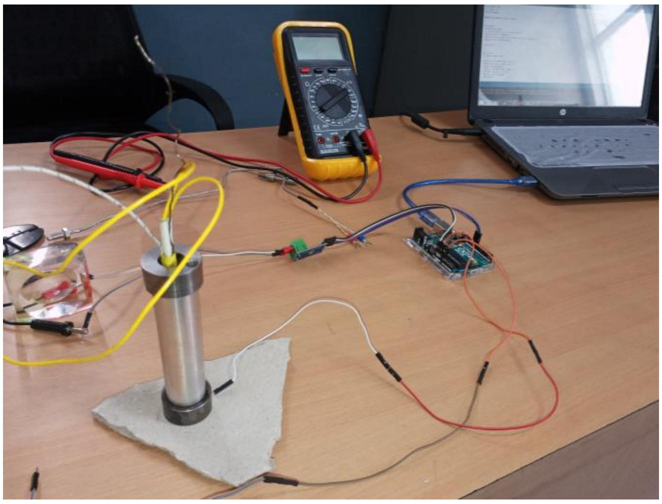


Fig. 6.2. Experimental Test Setup Image

The Fig 6.2 shows experimental test setup of thermionic energy conversion. The Arduino Uno is programed using MAX6675 module for measuring emitter temperature and A_0 is used to measure voltage between emitter and collector.

6.1. General Description of MAX6675

The MAX6675 compensates for cold junctions and digitises a type-K thermocouple signal. The data is output in a read-only format with a 12-bit resolution and SPITM compatibility. This converter has a resolution of 0.25°C and can read temperatures up to +1024°C.

6.2. Arduino Uno program

```
#include "max6675.h"
 int ktcSO = 8;
int ktcCS = 9;
int ktcCLK = 10;
int value = 0;
float voltage;
 MAX6675 ktc(ktcCLK, ktcCS, ktcSO);
 void setup() {
  Serial.begin(9600);
  // give the MAX a little time to settle
  Serial.println("CLEARDATA");
  Serial.println("LABEL,Time,Started Time,Deg C,Voltage,");
  Serial.println("RESETTIMER");
  delay(500);
}
 void loop() {
  // basic readout test
   Serial.print("DATA,TIME,TIMER,");
   Serial.print(ktc.readCelsius());
   value =analogRead(A0);
   Serial.print("DATA,TIME,TIMER,");
   voltage = value * 5.0/1024;
   Serial.println(voltage);
   delay(500);
}
```

6.3. Modifications in Test Setup

Experiments were conducted on thermionic conversion test setup using tungsten as an emitter and molybdenum as an emitter. Tungsten and molybdenum are the low work function metals and have higher melting point. The experiment included heating of emitter with cartridge here and collecting the live data from TEC test setup for emitter temperature (^{0}C) and potential difference (V). The tasks performed during experimentation are elaborated below;

6.3.1 Clamping design, fabrication and Installation

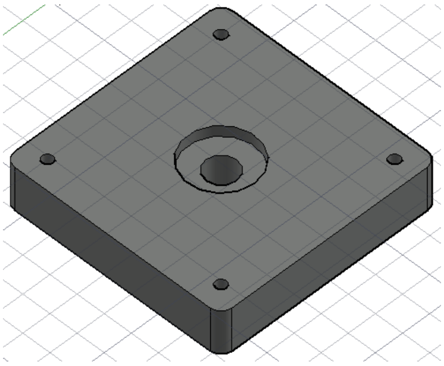


Fig. 6.3. Design of clamp   Fig. 6.4. Test setup assembly with clamp

Thermionic conversion test setup is modified with clamps. The clamp design and its assembly is shown in Fig. 6.3. & Fig. 6.4. It was manufactured in school of manufacturing skills on conventional milling machine.

6.4. Material Selection

Emitter and collector materials must have good thermionic emission characteristics. The efficiency of TEC is determined by the work function of the material. Work function is a material property that indicates a material's ability to emit electrons at a specific temperature. It represents the amount of energy necessary for a metal to emit electrons. The temperature at which electrons are released decreases as the work function decreases. Different materials capable of thermionic emission are considered based on the available literature. The materials and their properties are mentioned in Table 6.1.

Table 6.1. High Work Function Materials

Material	**Work Function (eV)**	**A (A/cm2)**	**Melting Point (^{0}C)**
Mild Steel (MS)	4.81	120	1510
Aluminium (Al)	4.26	120	660
Brass	5.10	80	940
Tungsten (W)	4.54	60	3422
Molybdenum (Mo)	4.15	55	2623
Beryllium (Be)	4.98	65	1287

Low work-function materials capable of solid-state thermionic conversion are listed in Table 6.2. When compared to non-coated emissive materials, the study on these materials demonstrated that coating them can achieve ultra-low work-functions, lowering the temperature of emission.

Table 6.2. Low Work Function Coating Materials

Coating Material	**Work Function (eV)**
Graphene	1.1
CsI	2.6
Carbon nanotube	1.4
Diamond	0.9

From the Table 6.1 & 6.2 it can be stated that, the materials with low work function value and higher melting point must be used as emitter and collector, to obtain higher thermionic conversion efficiencies.

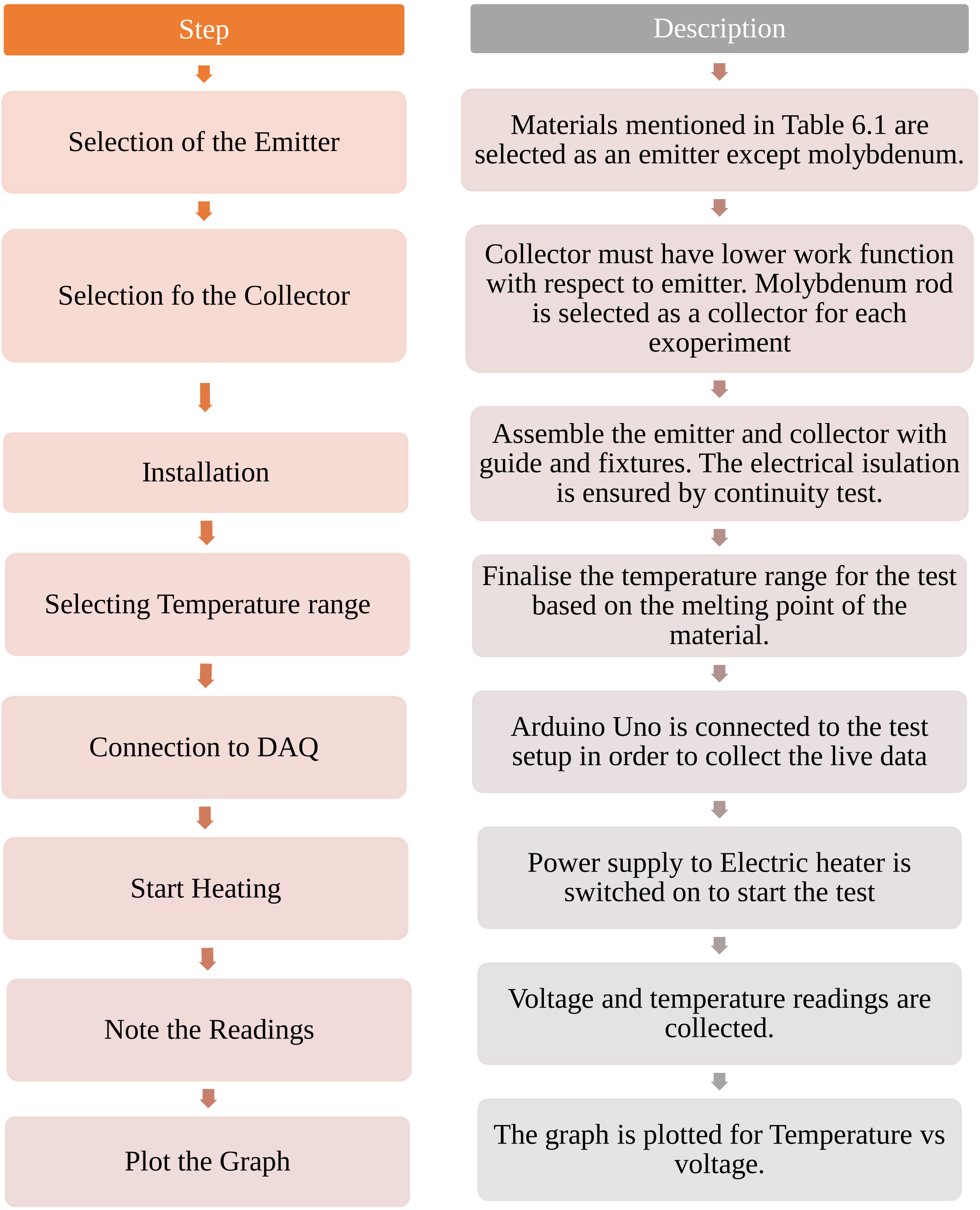
6.5. Assumptions

1. The emitter and collector are electrically & thermally insulated.
2. Temperature measurements are carried out using K type thermocouple.
3. The emitter is heated with an electric cartridge heater in order to obtain its emission temperature.
4. The temperature range of emitter is 30^{0}C to 800^{0}C. Whereas for Aluminium and brass the temperature is kept 450^{0}C due to their lower melting point.
5. The experiments are performed under controlled environment where the dynamic characteristics of exhaust pipe are not created.

6. As collector must have lower work function than emitter. Hence for all experimentation, molybdenum electrode is used as a collector.

6.6. Experimentation Process

The objective of experiment is to identify the performance of materials for thermionic emission. The small scale thermionic regenerator has two electrodes emitter and collector. The materials shown in Table 6.1 are considered for experimentation. The details of their work function, Richardson constant and melting point are mentioned. The steps shown in Fig 6.5 below are performed during experiment;



Step	Description
Selection of the Emitter	Materials mentioned in Table 6.1 are selected as an emitter except molybdenum.
Selection fo the Collector	Collector must have lower work function with respect to emitter. Molybdenum rod is selected as a collector for each exoperiment
Installation	Assemble the emitter and collector with guide and fixtures. The electrical isulation is ensured by continuity test.
Selecting Temperature range	Finalise the temperature range for the test based on the melting point of the material.
Connection to DAQ	Arduino Uno is connected to the test setup in order to collect the live data
Start Heating	Power supply to Electric heater is switched on to start the test
Note the Readings	Voltage and temperature readings are collected.
Plot the Graph	The graph is plotted for Temperature vs voltage.

Fig. 6.5. Experimental Process

The same process is repeated for all selected emitters and findings are discussed in results chapter. After these experimentations the results were compared with the simulation results and the findings are discussed in the results chapter.

Without limiting to test setup experimentations, this study is extended to evaluate the effectiveness of a TRS on HEV performance. This study is done by modelling and simulation of a full scale hybrid electric vehicle, which is explained in following sections;

6.7. Full Scale Vehicle Modelling & Simulation

The thermionic regeneration system simulations are done according to prototype and test setup environment. The results may vary when the system is installed on vehicle. Also the system must work same during the vehicle run. Therefore it becomes necessary to simulate the system model collaborated with a full vehicle model. This will provide an overall and exact behaviour of a designed system.

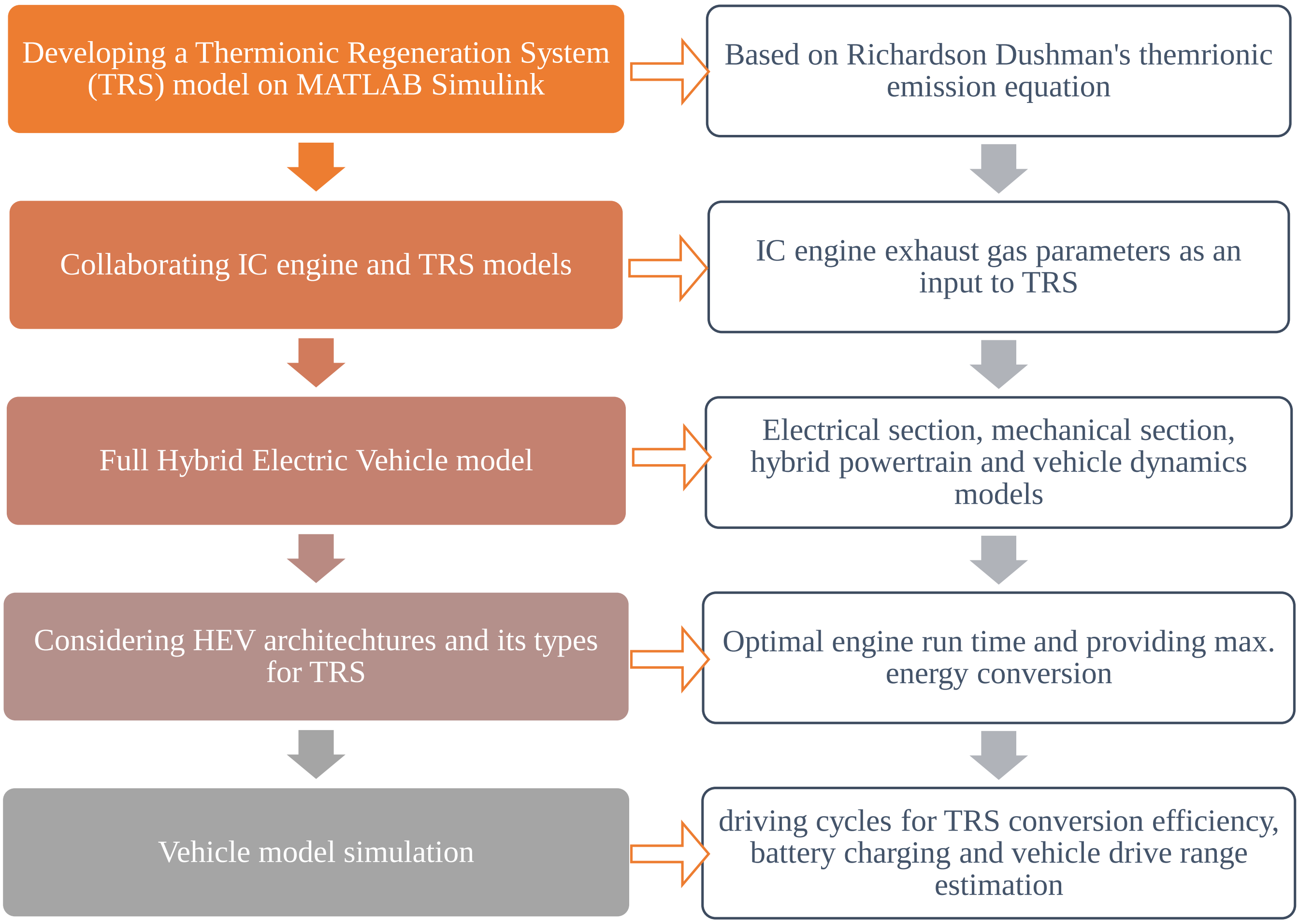


Fig. 6.6. Steps for Performing Full Vehicle Simulation with TRS on MATLAB Simulink

The approach shown in Fig. 6.6 is used for full vehicle simulation.

Automotive vehicle simulation is done to analyse vehicle performance in terms of acceleration, engine speed vs. torque and fuel consumption. Thermionic regeneration system is designed to suit hybrid electric vehicle electrical system. MATLAB Simulink provides a math environment to model different hybrid electric vehicle systems, its combinations and architectures.

6.8. Developing a Thermionic Regeneration System (TRS) Model

The block diagram shown in Fig. 6.7 expresses the integration of a thermionic regeneration system model with hybrid electric vehicle model (HEV) in Simulink. Normally, HEV is divided into electrical, mechanical, powertrain and vehicle dynamics sections.

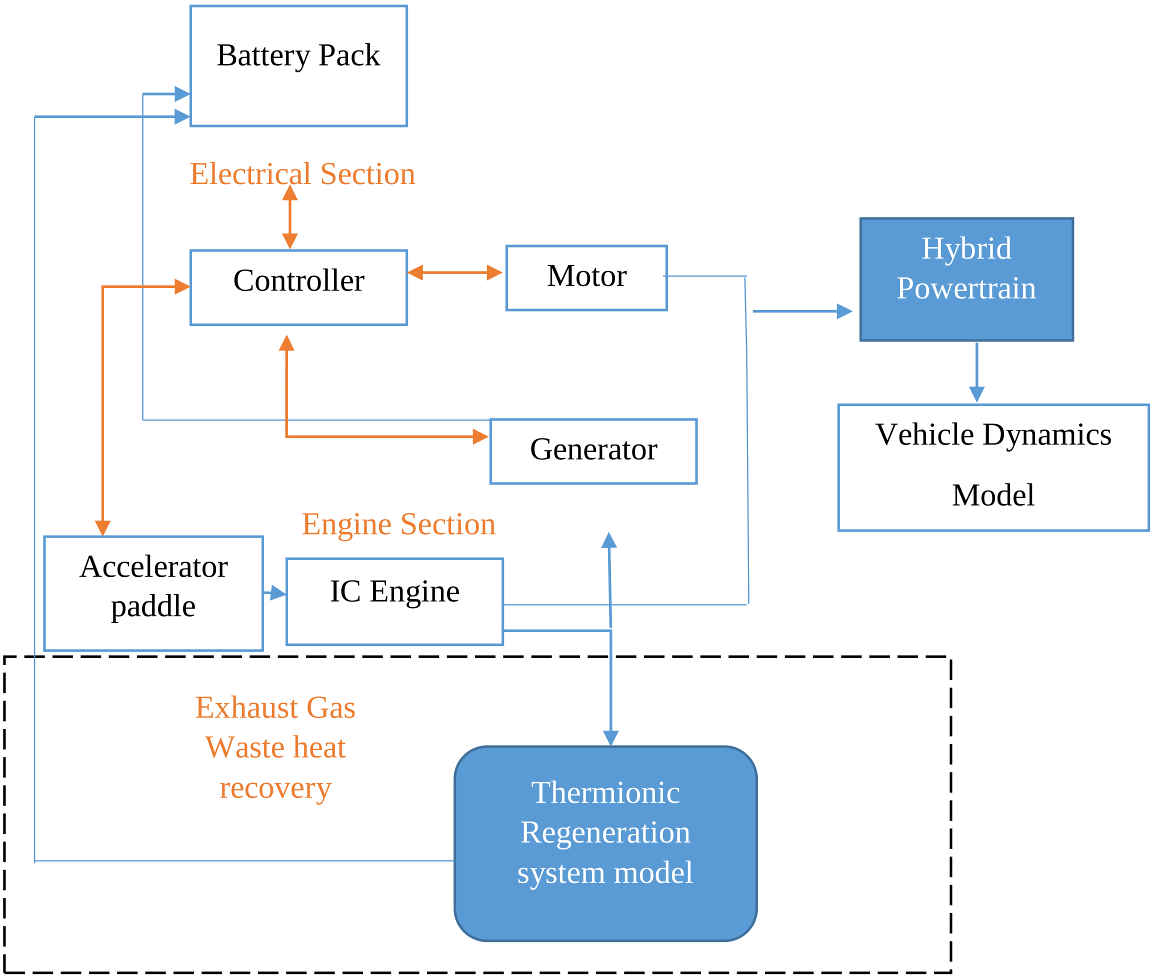


Fig. 6.7. Simulink Models and their Collaboration (block diagram)

TRS is a math model representing the conversion rate of heat energy to electrical power. Exhaust gas temperature is an input to TRS, taken from IC engine model. The output of TRS is an electric power recovered from waste heat and is supplied to battery for storage. The controller shifts the vehicle driving modes from mechanical to electric as per the algorithm. As TRS is add on system the algorithm needs a slight modification so that the fuel consumption will be further reduced.

Thermionic regeneration system model is developed using Simulink math functions as shown in Fig. 6.8. The input parameters for convertor are IC engine exhaust gas temperature and mass flow rate. The output of this system is current density and voltage plots. The numerical values of voltage and current show the conversion rate. Further the system is connected to battery model same as generator for battery charging.

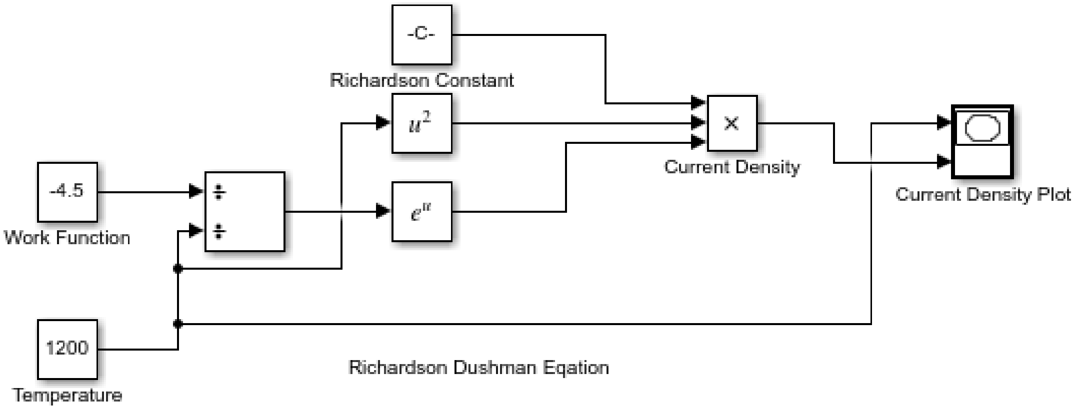


Fig. 6.8.Thermionic Regeneration System (TRS) Model *(Courtesy: SIMULINK)*

The model variables like work function and thermionic emission properties can be changed to consider different materials for the analysis. The tungsten as emitter and molybdenum as collector have given better conversion efficiency compared to other conventionally used exhaust pipe materials. Therefore, in this analysis the thermionic regeneration system is considered with tungsten and molybdenum only. The simulations using graphene and low work function coatings have shown much better results.

This system level model is further collaborated with an existing IC engine model. The IC engine model is available in powertrain block set.

6.9. Collaborating IC engine and TRS model

Simulink have different IC engine models in the application library expressing the engine performance and its mechanical functions. The images shown in Fig. 6.9 are the models considered for this study. The following models are considered because these models have engine exhaust parameters as an output.

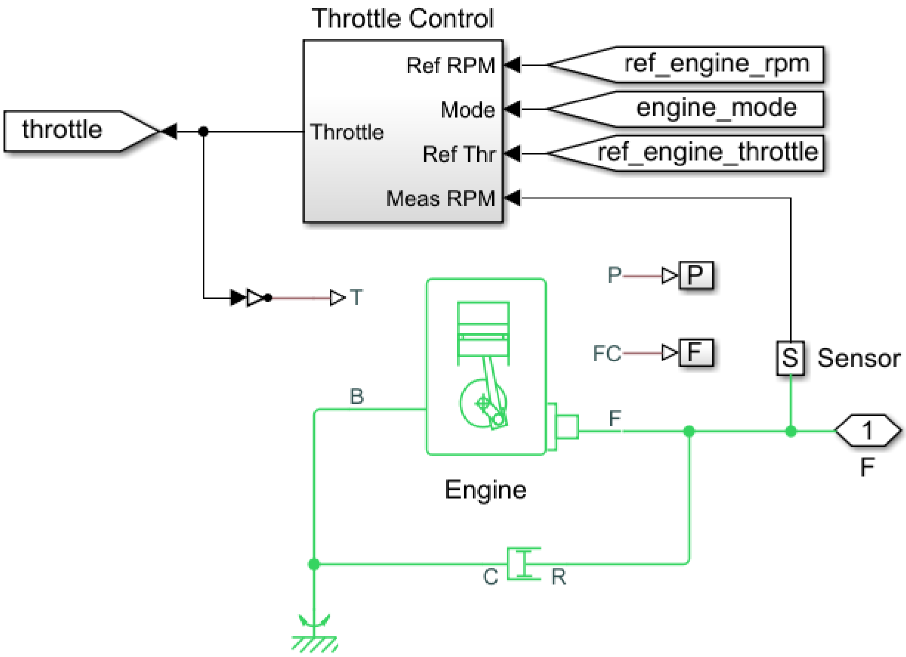


Fig. 6.9. IC Engine with Throttle and Rotational Inertia and Time lag

Source: (*Internal Combustion Engine with Throttle and Rotational Inertia and Time Lag - MATLAB - MathWorks India*, n.d.)

A generic internal combustion engine is represented by the Generic Engine block. Spark-ignition and diesel engines are examples of engine types. Parameterizations for speed-power and speed-torque are available. The normalised engine torque is specified via a throttle physical signal input. Crankshaft inertia and response time lag are two dynamic factors that can be selected. Based on the fuel consumption model selected, a physical signal port outputs the engine fuel consumption rate. Engine stall prevention and cruise control are possible with optional speed and redline controllers. The Generic Engine model uses a pre-programmed torque-speed relationship that is adjusted by the throttle signal by default.

- The engine model is specified by an engine power demand function g (Ω).

For a particular engine speed, the function returns the maximum power available. This function is normalised to physical maximum torque and speed values by the block parameters (maximum power, speed at maximum power, and maximum speed). The real engine power is specified by the normalised throttle input signal T.

In a steady condition at a set engine speed, the power is delivered as a fraction of the maximum power possible. It modulates the actual power delivered, P, from the engine:

$P(\Omega, T) = T*g(\Omega)$.

- The engine torque is $\tau = P/\Omega$.

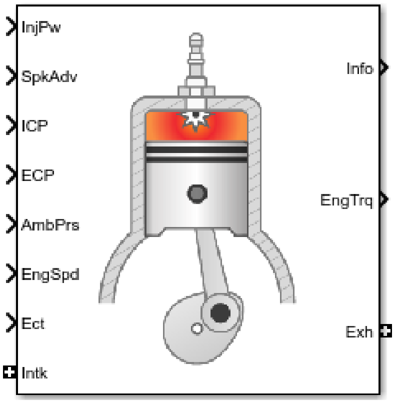


Fig. 6.10. SI and CI Core Engine

Source: (*Spark-Ignition Engine from Intake to Exhaust Port - Simulink - MathWorks India*, n.d.) (*Compression-Ignition Engine from Intake to Exhaust Port - Simulink - MathWorks India*, n.d.)

The SI and CI Core Engine block shown in Fig. 6.10 incorporates a spark-ignition (SI) and compression-ignition (CI) engine from the intake to the exhaust port. Larger vehicle models, hardware-in-the-loop (HIL) engine control design, and vehicle-level fuel economy and performance simulations can all benefit from the block.

- The Core Engine block calculates:

1. Brake torque
2. Fuel flow
3. Port gas mass flow, including exhaust gas recirculation (EGR)
4. Air-fuel ratio (AFR)
5. Exhaust temperature and exhaust mass flow rate
6. Engine-out (EO) exhaust emissions

7. Hydrocarbon (HC)
8. Carbon monoxide (CO)
9. Nitric oxide and nitrogen dioxide (Nox)
10. Carbon dioxide (CO_2)
11. Particulate matter (PM)

Exhaust calculations in engine model: The block calculates the;

- Exhaust gas temperature
- Exhaust gas-specific enthalpy
- Exhaust gas mass flow rate
- Engine-out (EO) exhaust emissions:

The exhaust temperature determines the specific enthalpy.

$$h_{exh} = CP_{exh} \times T_{exh} \ldots (1)$$

The exhaust mass flow rate is the sum of the intake port air mass flow and the fuel mass flow.

$$m_{exh} = m_{intake} + m_{fuel} \ldots (2)$$

To calculate the exhaust emissions, the block multiplies the emission mass fraction by the exhaust mass flow rate. To determine the emission mass fractions, the block uses lookup tables that are functions of the engine torque and speed.

$$y_{exh,i} = f_{i\_frac}(T_{brake}, N) \ldots (3)$$

$$m_{exh,i} = m_{exh} \times y_{exh} \ldots (4)$$

The fraction of air and fuel entering the intake port, injected fuel, and stoichiometric AFR determine the air mass fraction that exits the exhaust.

$$y_{exh,air} = max \left[y_{in,air} - \frac{m_{fuel} + y_{in,fuel} m_{intake}}{m_{fuel} + m_{intake}} AFR_s \right] \ldots (5)$$

If the engine is operating at the stoichiometric or fuel rich AFR, no air exits the exhaust. Unburned hydrocarbons and burned gas comprise the remainder of the exhaust gas. This equation determines the exhaust burned gas mass fraction.

$$y_{exh,b} = max \left[\left(1 - y_{exh,air} - y_{exh,HC}\right), 0 \right] \ldots (6)$$

- The above equations use the variables listed in Table 6.3.;

Table 6.3. Variables for IC engine model

T_{exh}	Engine exhaust temperature
h_{exh}	Exhaust manifold inlet-specific enthalpy
Cp_{exh}	Exhaust gas specific heat
$\dot{m}_{intk}$	Intake port air mass flow rate
$\dot{m}_{fuel}$	Fuel mass flow rate
$\dot{m}_{exh}$	Exhaust mass flow rate
$y_{in,fuel}$	Intake fuel mass fraction
$y_{exh,i}$	Exhaust mass fraction for i = CO_2, CO, HC, Nox, air, burned gas, and PM
$\dot{m}_{exh,i}$	Exhaust mass flow rate for i = CO_2, CO, HC, Nox, air, burned gas, and PM
T_{brake}	Engine brake torque
N	Engine speed
$y_{exh,air}$	Exhaust air mass fraction
$y_{exh,b}$	Exhaust air burned mass fraction

6.10. HEV Series Parallel Model

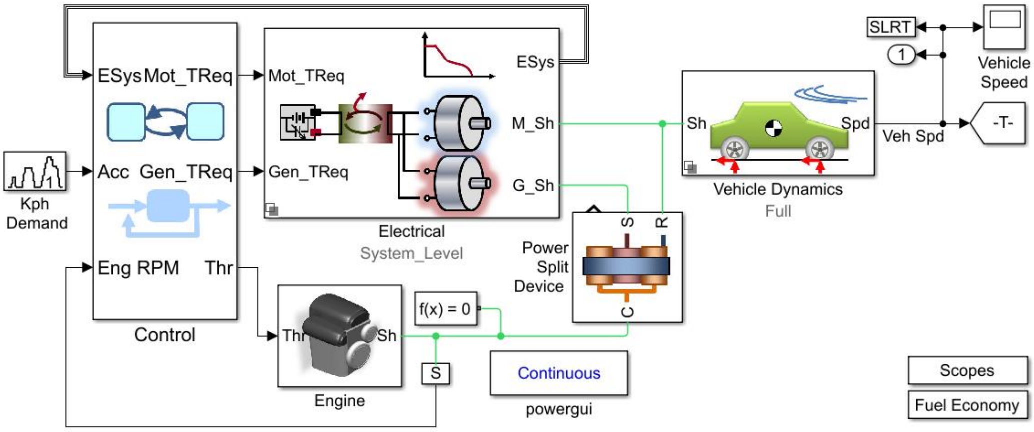


Fig. 6.11. Existing Model *(Courtesy: SIMULINK Matlab)*

The Fig. 6.11. shows existing hybrid electric model developed using Matlab Simulink. This has Control system, Electrical system, IC engine, power spit system & Vehicle dynamics system level models connected with each other for simulating vehicle performance. The input to all systems is given by driving cycles through Kmph Demand block. The vehicle performance parameters likes speed, fuel consumption & range are measured through scope & fuel consumption blocks.

6.11. Electrical System Level Modifications

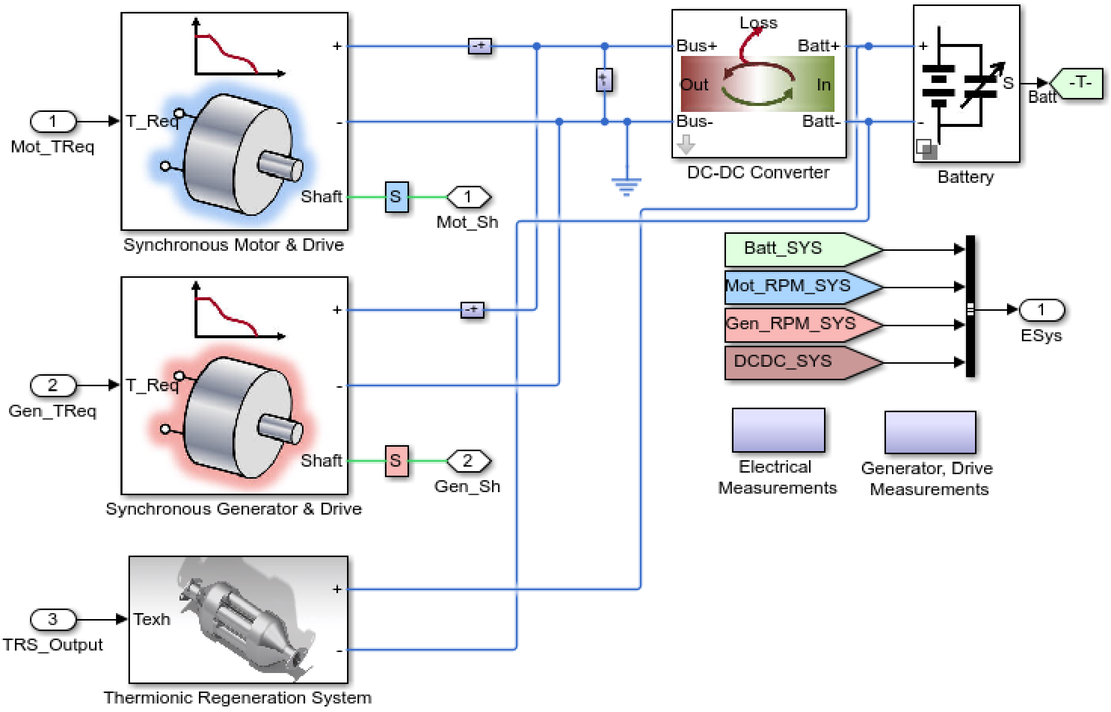


Fig. 6.12. Electrical System *(Courtesy: SIMULINK Matlab)*

The research proposes a thermionic regenerator installed in exhaust system of hybrid electric vehicle. Whereas the regeneration system is modelled here in the electrical system model. The output of TRS is connected to the battery as shown in the Fig 6.12. In addition to the existing motor and generator TRS is also included & electrical system is modified.

6.12. Structure of proposed algorithm

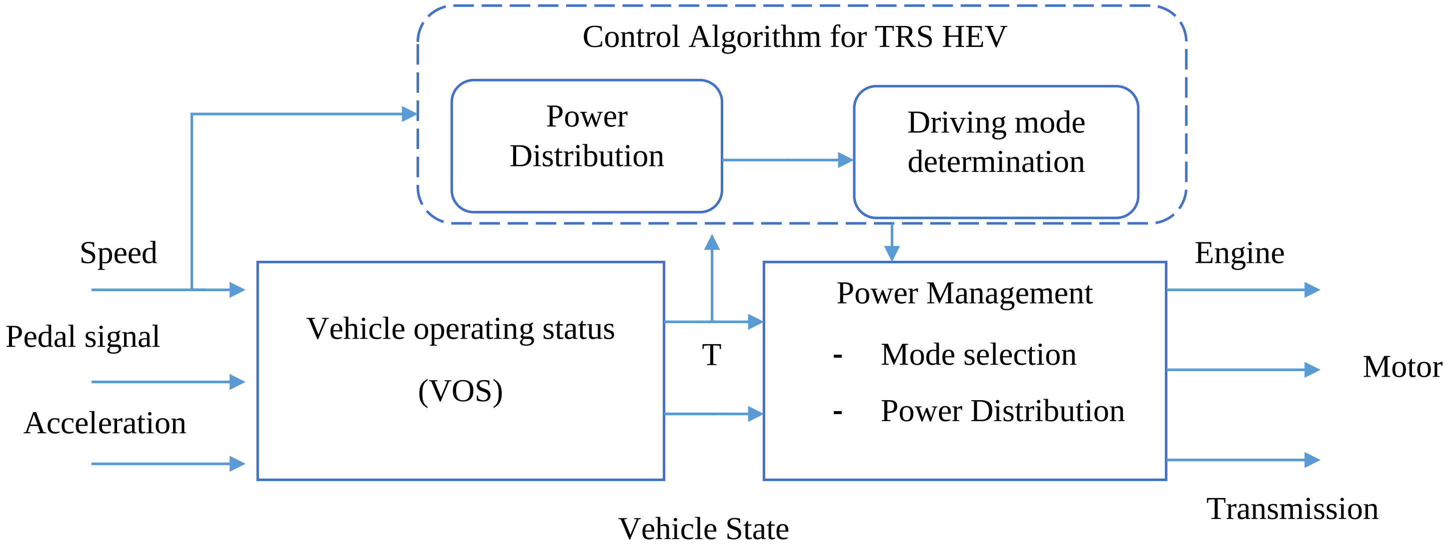


Fig. 6.13. Entire Structure of the Optimal Control Algorithm for TRS in HEV

To control the IC engine and EV mode of a HEV a control algorithm developed. The Fig 6.13 shows the entire structure of the optimal control algorithm for TRS in HEV. Based upon the vehicle's operating status (VOS) the power distribution and driving mode is determined. The control unit shifts the vehicles operation from IC engine to Electric as soon as Battery State of charge (SOC) is 100%.

The algorithm for waste heat regeneration and utilization is shown in following flow chart Fig. 6.14. When the vehicle operation is started either engine or electric mode is on. The TRS operation will start when the engine mode is on. The TRS output is measure and battery charging is started. The algorithm is implemented in the HEV model modified with TRS as shown in Fig. 6.15 using mode logic under control system block.

6.13. HEV algorithm for waste heat recovery using thermionic regeneration

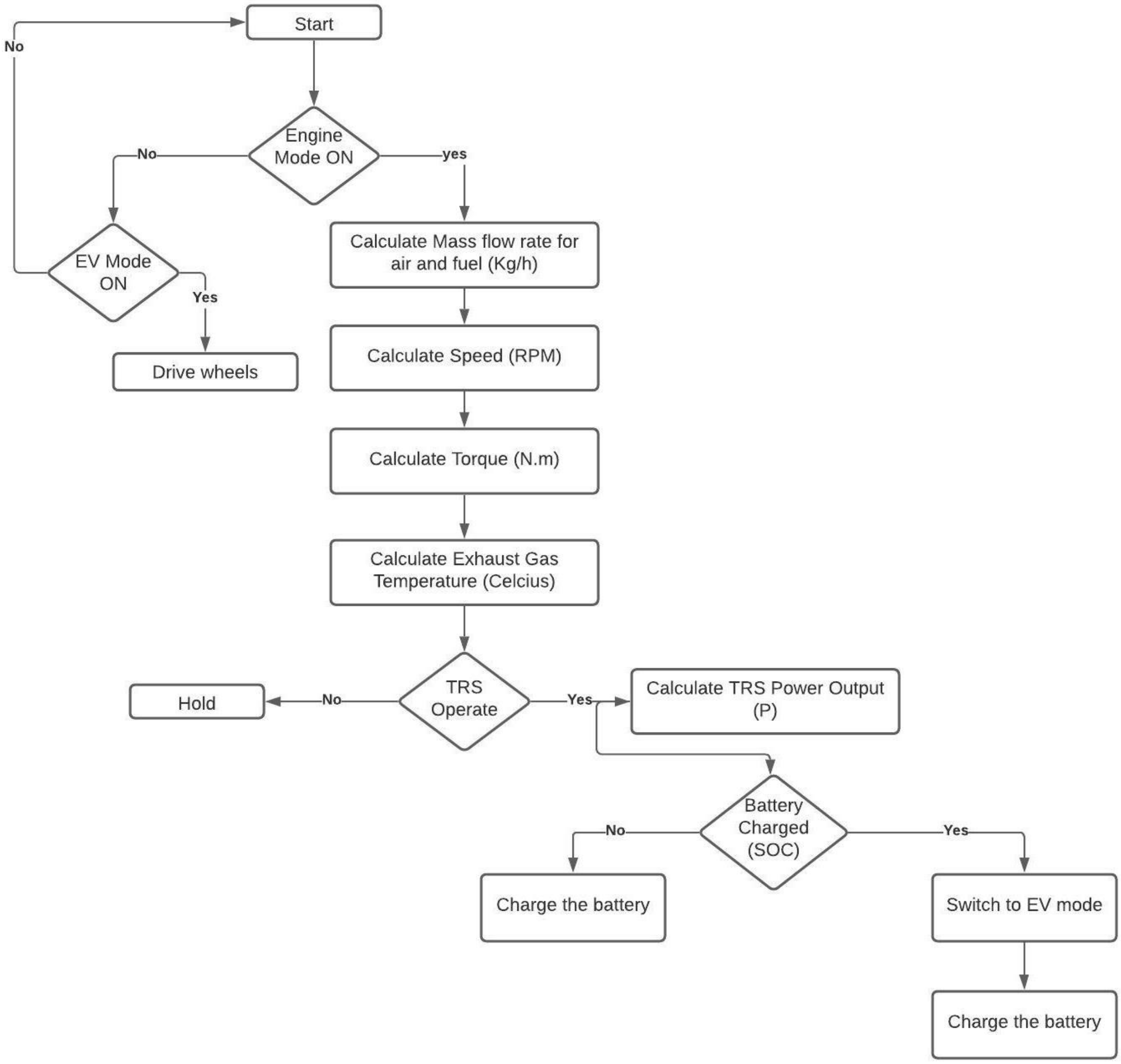


Fig. 6.14. TRS Control Algorithm

6.14. Implementation of TRS control algorithm in Model Logic under Control system

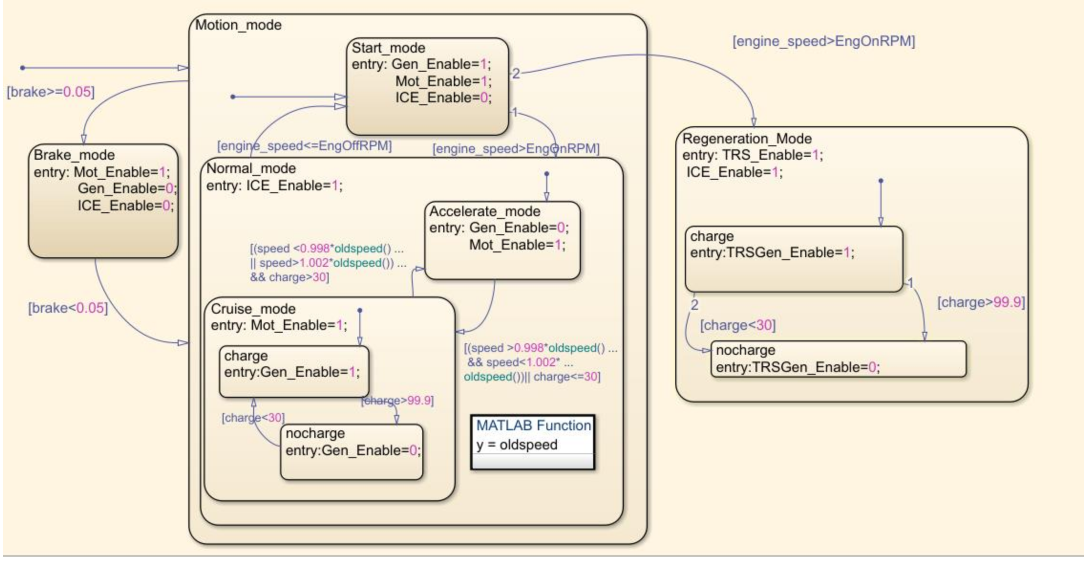


Fig. 6.15. TRS Control Algorithm implemented MATLAB Stateflow *(Courtesy: SIMULINK Matlab)*

6.15. Drive Cycles

Drive cycle represents the series of data points with vehicle speed in km/hr versus time in sec. Here the drive cycle is an input parameter for vehicle simulation. Following Indian Driving cycles are used for the analysis;

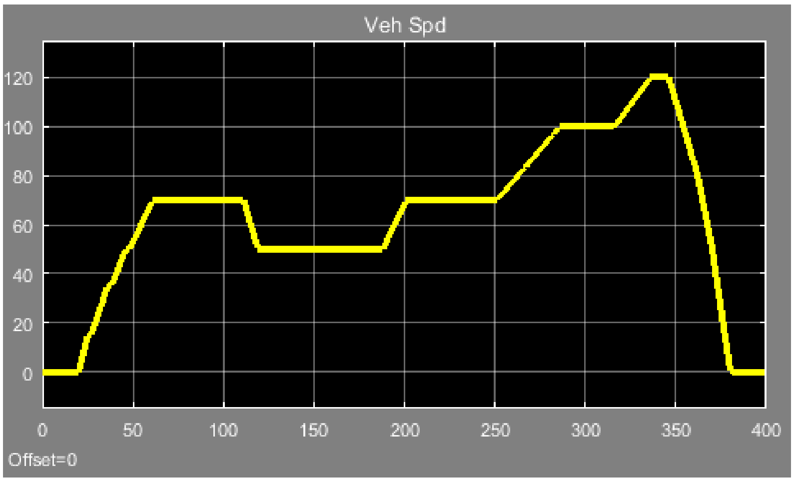


Fig. 6.16. Urban Drive Cycle 1 *(Courtesy: SIMULINK Matlab Library)*

Urban Drive Cycle 1: This drive cycle as shown in Fig 6.16 represents the highest vehicle speed of 120 Km/hr with gradual increase from 0 km/ hr. the total drive cycle run time is 400 Sec. This drive cycle represents on road/ highway operation of a vehicle.

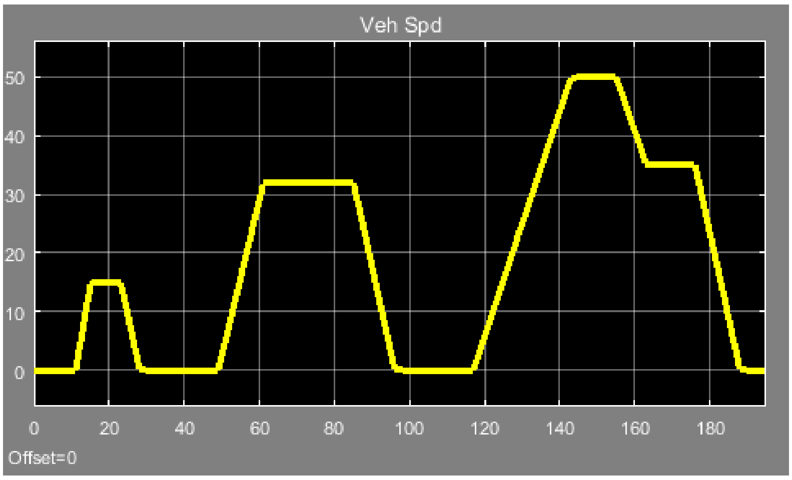


Fig. 6.17. Urban Drive Cycle 2 *(Courtesy: SIMULINK Matlab Library)*

Urban Drive Cycle 2: This drive cycle as shown in Fig 6.17 represents the highest vehicle speed of 0 to 50 Km/hr. The total drive cycle run time is 190 Sec. This drive cycle represents continuous acceleration & braking simulating traffic intercity operation.

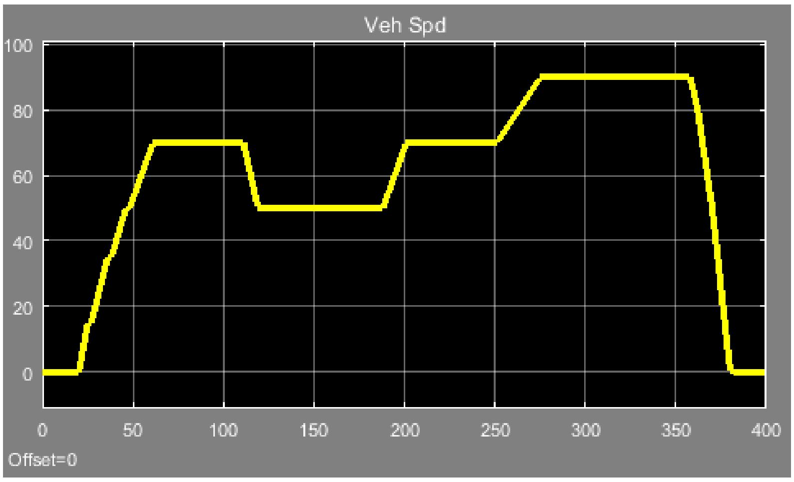


Fig. 6.18. Urban Drive Cycle 3 *(Courtesy: SIMULINK Matlab Library)*

Urban Drive Cycle 3: This drive cycle as shown in Fig 6.18 represents the highest vehicle speed of 0 to 90 Km/hr. The total drive cycle run time is 400 Sec. This drive cycle represents sudden increase in vehicle speed simulating fast acceleration. Thus, all the operating conditions of a HEV vehicle are being considered for this simulation.

6.16. TRS integrated HEV model

This model in the MATLAB library depicts the basic architecture of a two-mode hybrid transmission. It's been complied with and combined with TRS. Three planetary gear sets and four clutches make up the system. This setup allows for four fixed gear ratios as well as two power-split options. The power split modes are used to move between fixed gear ratios and to accelerate and decelerate quickly. When cruising, the fixed ratios aid efficiency. Only Clutch 1 is used in the first power split (input-split regime). Only Clutch 2 is used in the second power split (compound-split regime). When two clutches are engaged at the same time, one degree of freedom is removed, resulting in a fixed ratio.

As seen in Fig. 6.19, TRS is linked to the IC engine model. Engine and vehicle performance, as well as TRS power output, are plotted in the simulation charts. The Strategy subsystem, in this case, defines a set of rules for determining when to switch transmission modes during a continuous acceleration profile. Controlling electrical power as a function of speed, driver demand, and battery state would be part of a comprehensive vehicle control strategy.

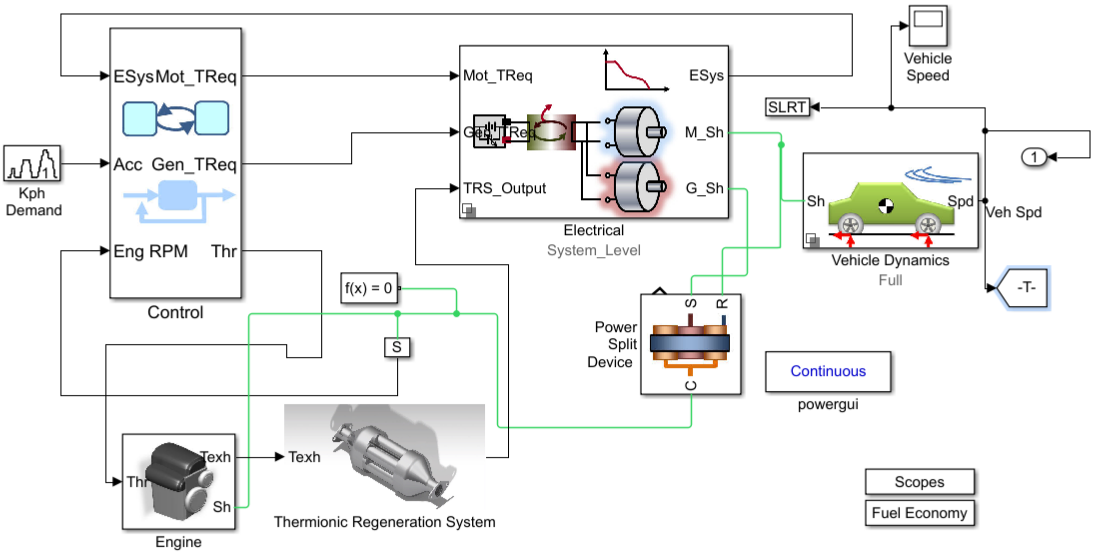


Fig. 6.19. TRS integrated HEV model

Source: (*Two Mode Hybrid Transmission - MATLAB & Simulink - MathWorks India*, n.d.)