Modelling of Electric Vehicle Dynamics using dSpace DS1103

Tomáš Haubert [1], Tomáš Hlinovský [1] and Pavel Mindl [1]
[1] Czech Technical University in Prague, Faculty of Electrical Engineering, Department of Electric Drives and Traction, Prague, Czech Republic, e-mail: *tomas.haubert@fel.cvut.cz*

Abstract — **This article deals with a dynamic model of the electric vehicle. The model is created in the MATLAB/Simulink and respects electric drive efficiency, traction battery charging/discharging efficiency and limited battery capacity. The model is prepared for uploading into the dSpace DS1103 development hardware and it creates at the same time the real-time system for modelling the electric vehicle dynamics. This system will allow connection with the 80 kW dynamometer for the real energy flow in the electric vehicle. This system will be a part of a complex system for the electric vehicle optimal control.**

Keywords — Electric vehicle, optimal control, modelling, MATLAB/Simulink.

I. INTRODUCTION

Increasing of living standards increased needs for people transport. Gas emissions and fossil fuel consumption are actual problems for global environment quality on the Earth. Gas emission of car internal combustion engines (ICE) brings many ecological problems in big cities specially [1].

Electric vehicles are tackling this problem, because they do not produce gas emissions. Electric energy needed on the board can be stored in batteries, but their disadvantages are limited capacity and inconvenient ratio of capacity to mass. Driving area of electric vehicles (EV) is lower in comparison with vehicles equipped by ICE.

II. BASIC DYNAMIC MODEL OF EV IN THE MATLAB/SIMULINK

A. Basic equations, relations and variables

$$F_{acc} = m_{red} * a(t) \tag{1}$$

$$F_{air} = \frac{1}{2}\rho_a * A * C_d * v^2(t) \tag{2}$$

$$F_{roll} = c_r * m * g * \cos(\alpha(s)) \tag{3}$$

$$F_{grade} = m * g * \sin(\alpha(s)) \tag{4}$$

m	… weight of EV
$a(t)$	… acceleration of EV
ρ_a	… air density
A	… frontal surface
C_d	… aerodynamic constant
$v(t)$	… speed of EV
c_r	… rolling drag force coefficient
g	… gravity constant
$\alpha(s)$	… slope (function of trajectory)

According to the general relation for power:

$$P(t) = F * v(t) \tag{5}$$

it is possible to create the following power balance equation:

$$P(t) = \left(F_{acc} + F_{air} + F_{roll} + F_{grade}\right) * v(t)\ . \tag{6}$$

B. Description of the dynamic model subsystems

According to these above mentioned equations subsystem the Vehicle Dynamic Model has been created. The inputs

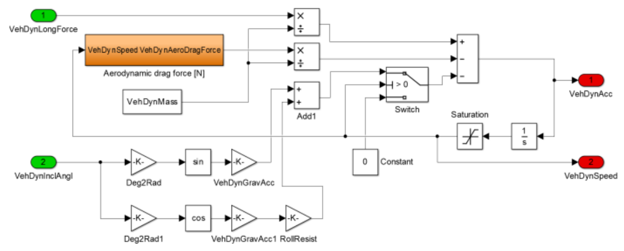

Fig. 1 Subsystem Vehicle Dynamic Model

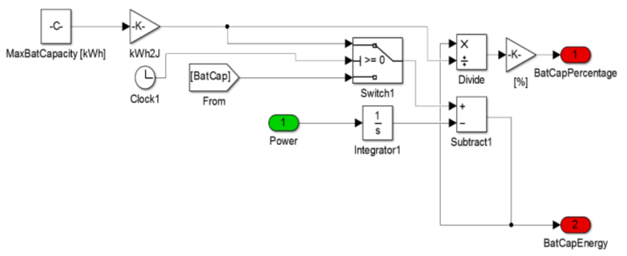

Fig. 2 Subsystem Energy Consumption

of the subsystem are forward traction force (VehDynLongForce) and track slope (VehDynAngl) and outputs are the acceleration of the EV (VehDynAcc) and speed (VehDynSpeed), which is calculated as an integration of acceleration. A next subsystem is named Energy Consumption. In this subsystem, we add or subtract the consumed or added (recuperated) energy into the stack of energy which is represented as an accumulator. The outputs are just the current state of charge (in percent value) and a value of the remaining energy stored in the battery.

The model can also calculate with an efficiency map of the electric motor. Because of the absence of the synchronous motor with permanent magnets model, it is necessary to get the efficiency from the motor map. The speed of the EV is known and we have a defined type of tyres with specific dimensions. Then we can calculate the torque generated by the motor on the shaft. The subsystem EMotor_Calculation has two inputs, speed of the EV and current power generated by the motor.

Revolutions per minute of the electric motor are calculated as:

$$n_{mot} = \frac{v}{3{,}6 * TyrePerimeter} * GearRedRatio * 60 * v(t) \quad (7)$$

Variables TyrePerimeter and GearRedRatio will be explained later. The torque generated by the electric motor is calculated as:

$$M_{mot} = \frac{P}{n_{mot}} * \frac{60}{2\pi} \quad (8)$$

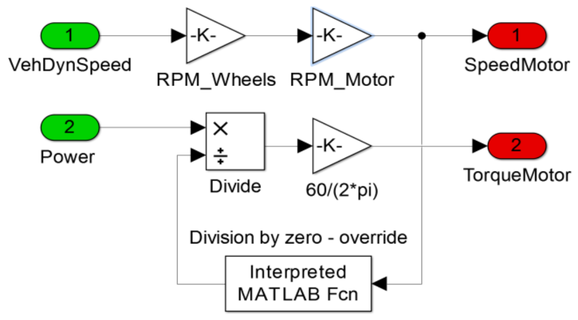

Fig. 3 Subsystem EMotor_Calculation

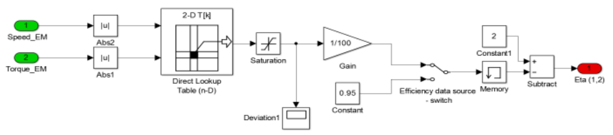

Fig. 4 Subsystem Efficiency_map

Efficiency map of the synchronous motor with permanent magnets is a graphical representation of the relation η (M, n). Since M (torque) and n (speed of the electric motor) are known, the model can respect the actual position of the operating point in this map.

The data with the efficiency values are written in m-file, where it is stored with a certain accuracy and density of measured points (let this be just a general efficiency of a common synchronous motor with permanent magnets for our purpose.). Block Direct Lookup table can perform an interpolation, so it is able to find a value of efficiency for every motor speed in the given range. The subtraction (2-eff) has to be there, because the model multiplies then the value of power, needed for the motion of the EV, with this number and the power consumed from the battery must be greater than the power generated by the electric motor. So it is necessary to divide the value of efficiency, which lies in the interval (0, 1), by number 1.

C. Completion of the model and initialization files

So the model as a whole has two basic inputs: speed of EV and slope of the track. The forward traction force is represented as a tuned PI controller for now. The input of the PI controller is the reference speed and output is the above mentioned traction force. The other blocks surrounding the subsystems are mainly for unit conversions or logic of charging/discharging the battery. For testing of the model, we used basic drive cycles ECE and NEDC. The ECE cycle represents a drive in city traffic conditions and NEDC cycle consists of 3 ECE cycles plus one cycle representing drive on a highway, expressway or city bypass. Data of these drive cycles consists of speed in dependence on time. This is the reference speed for the PI controller. Furthermore it is needed to run the m-file with the efficiency data. The main file is the initialization file (below):

```
% Init m-file for VehDyn_model (version 5, 21/5/2014)

VehDynGravAcc = 9.81;

% Vehicle parameters
VehDynMass = 1200; %[kg]
VehDynAeroDragCoef = 0.3;
VehDynMassDensityOfAir = 1.3; %[kg/m^3]
VehDynFrontalArea = 2; %[m^2]
MaxBatCapacity = 7; %[kWh]
Cr = 0.01; % Rolling friction coeficient
% Tyre dimensions + Tyre perimeter calculation
TyreWidth = 185; %[mm]
TyreAspectRatio = 0.6; % [%] Height of the tire itself
= TireWidth*TireAspectRatio*0.01
DiscDiameter = 15; %[Inches] 1 inch = 2.54 cm
TyrePerimeter =
((TyreWidth*TyreAspectRatio)*2+TyreDiameter*25.4)*pi*0.
001*0.92; %[m]

%Drivetrain parameters
GearRedRatio = 4.5; % eg. Tesla Model S has a single
speed gear with 9.73/1 reduction ratio.
```

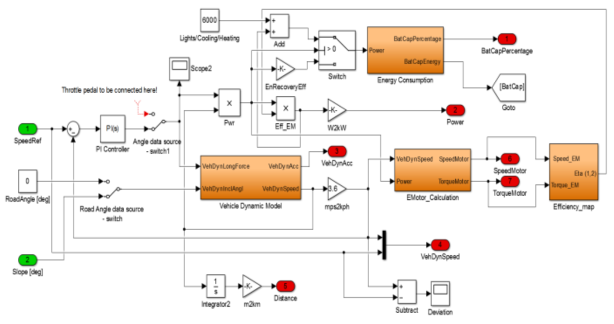

Fig. 5 Subsystems Vehicle Dynamic Model + Energy Consumption

The fixed transmission has been set on the value 4.5, because if we get the speed from relation (7), so at 7000 rpm the EV should be going forward with a speed of about 161 km per hour in the defined tyre configuration. Braking is only represented as a recuperation, so if the EV is braking, all the energy is getting stored in the battery with an efficiency of about 25 %. Whole model is put in one last subsystem (Fig. 6), so one can clearly see the inputs and outputs. One can also notice the preparation for an optimization of the whole drive and inclusion of the track slope profile. However the drive cycles ECE and NEDC do not respect sloping of the track.

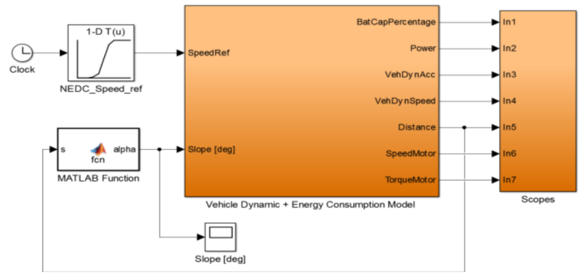

Fig. 6 Whole model with inputs and outputs

D. Results of the model using the NEDC driving cycle

Some basic results can bee seen below by using the speed and time data files of the NEDC driving cycle to run the model.

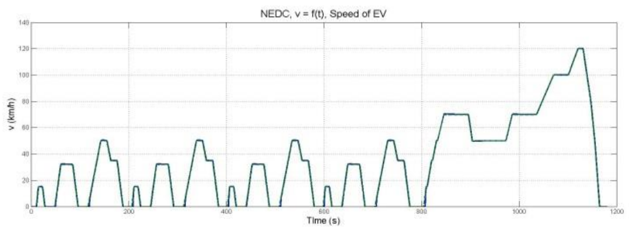

Fig. 7 NEDC: v = f(t), v_{ref}, v_{real} = f(t)

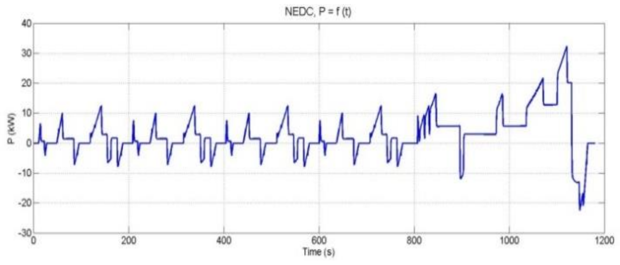

Fig. 8 NEDC: P_{mech} (kW) = f(t)

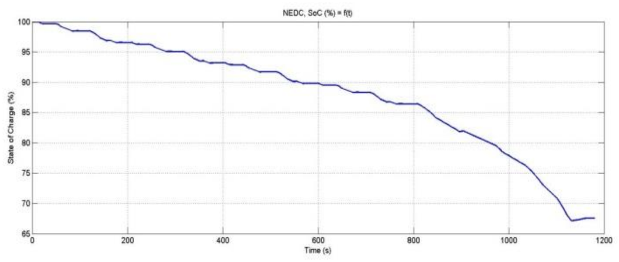

Fig. 9 NEDC: SoC (%) = f(t)

E. Results of the model using specific Artemis driving cycle

The Artemis-mw-130-incl-pre-post driving cycle has been also used to demonstrate a faster and longer lasting driving. Some of the results can be also seen in the graphs placed below this paragraph.

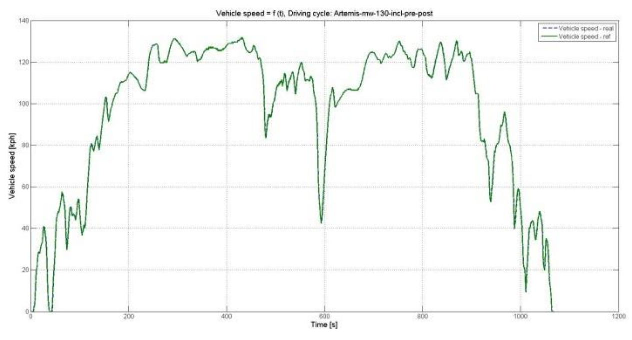

Fig. 10 Artemis: v_{ref}, v_{real} = f(t)

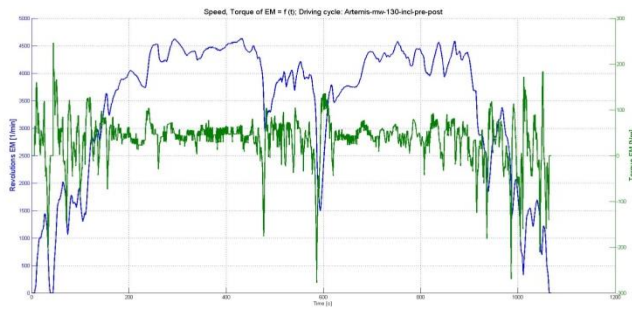

Fig. 11 Artemis: M_EM, n_EM = f(t)

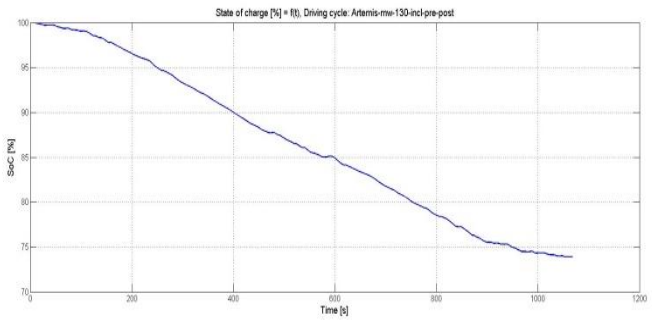

Fig. 12 Artemis: SoC (%) = f(t)

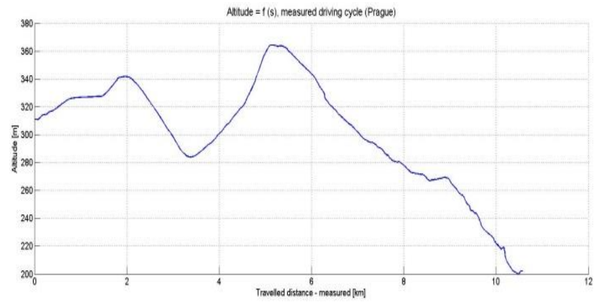

Figure 13: Real driving cycle - city, Altitude (m) = f(s)

III. RESULTS

In this paragraph we would like to present results of a real measured driving cycle in Prague traffic and on highway. Data of the route was taken by an app called myTracks which was run on iPhone 5S using the built-in GPS module. We used gasoil car for the measured driving cycle in Prague. The all data are from the built-in GPS module. The data file consists of current longitude, latitude, altitude and travelled distance between the current and before sampled point, absolute travelled distance and speed. Speed should be of course taken from the car system as a physically measured value. This includes then a small deviation which is reflected in a different real travelled distance and distance computed by the vehicle model in Simulink. The moving average for 20 samples was used for the altitude filtering. The goal is to repeat the same driving cycle 10 times for better altitude accuracy.

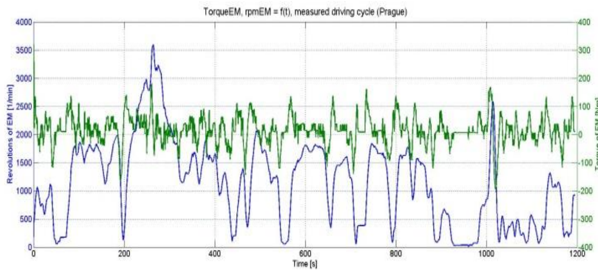

Figure 14: Real driving cycle - city, n_EM, M_EM = f(t)

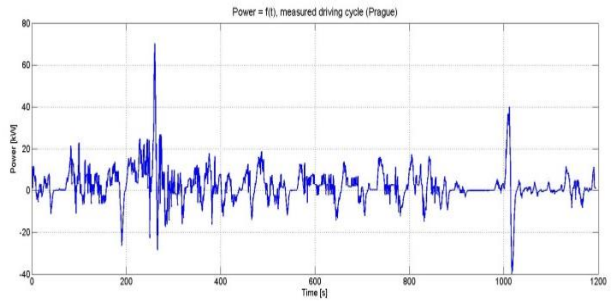

Figure 15: Real driving cycle - city, P(kW) = f(t)

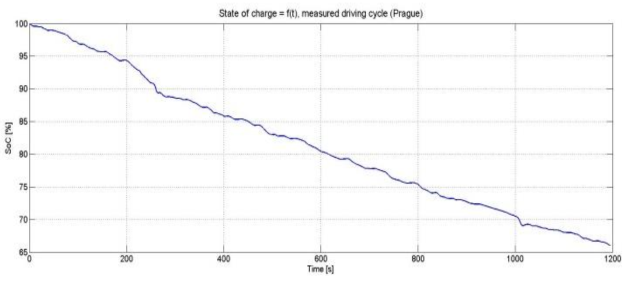

Figure 16: Real driving cycle - city, SoC (%) = f(t)

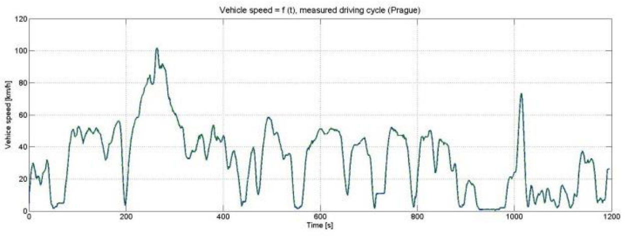

Figure 17: Real driving cycle - city, Vehicle speed (km/h) = f(t)

The real driving cycle – a highway with nearly 50 km of travelled distance had to be simulated with a greater battery capacity, which was set to 24 kWh. Previous simulation of the city cycle was simulated with default 7 kWh.

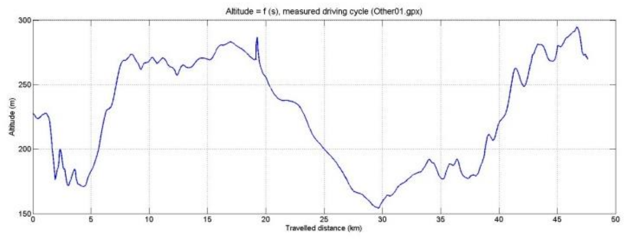

Figure 18: Real driving cycle - highway, Altitude (m) = f(s)

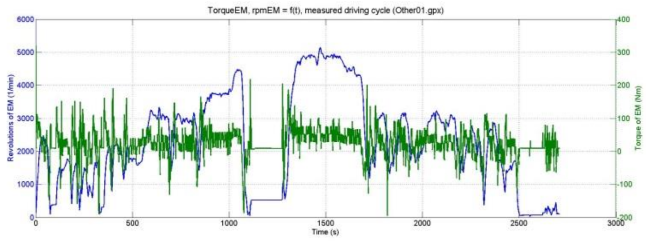

Figure 19: Real driving cycle - highway, n_EM, M_EM = f(t)

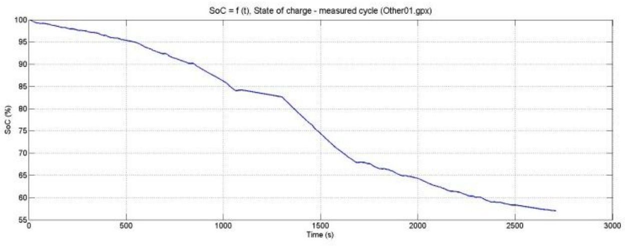

Figure 20: Real driving cycle - highway, SoC (%) = f(t)

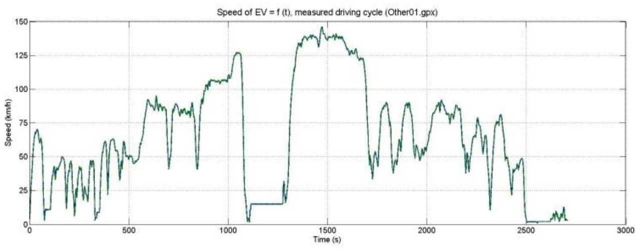

Figure 21: Real driving cycle - highway, Vehicle speed (km/h) = f(t)

IV. Hardware Structure

The dSpace DS1103 is a single-board system with real-time processor and comprehensive I/O. It is used for the modern rapid control prototyping and it should be used for induction motor control, automotive applications, robotics, etc.

Figure 22: dSpace AutoBox

The dSpace DS1103 board is included in an expansion box called AutoBox. The AutoBox is dedicated for using in a car. The AutoBox supply voltage is 12 V. The AutoBox has included an Autoboot option. This option enables the boot of application from the CF (Compact Flash) card without a computer. An interface board was designed for connection between the dSpace and master controller (MC) and dynamometer (DM). The DS1103 enables the real-time simulation of the EV and produces the torque which simulates the load torque in a real EV.

The communication between the DS1103 and dynamometer is realized via CAN. The dynamometer has own control and measurement unit and it is produced by VUES Brno Company. The communication between the DS1103 and MC is realized via the RS232 using own developed protocol. The diagram of the hardware structure is shown in Fig. 22. Permanent Magnet Synchronous Machine (PMSM) is used for the main electric drive of the electric vehicle. The power electronics is used for the PMSM control and communication with the MC is realized via CAN.

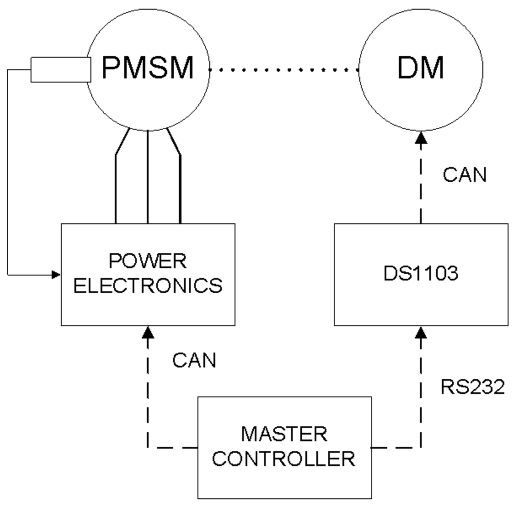

Figure 22: Hardware Structure of the Laboratory Stand

V. Conclusion

This article presents using of the dSpace DS1103 development system for the electric vehicle dynamics modeling. The model has two operation modes. The first one is the offline mode where the input is a driving cycle and the output are dynamic variables such as torque, speed, electric current, etc. The second one is online mode where the inputs are speed and the track condition and the output is torque for dynamometer. The connection between the DS1103 and dynamometer is missing for this mode and it is currently under development.

Acknowledgment

This research has been realized using the support of Technological Agency, Czech Republic, programme Centres of Competence, project TE01020020 Josef Bozek Competence Centre for Automotive Industry. This support is gratefully acknowledged.

References

[1] T. Haubert, J. Bauer, P. Mindl: Using of dSpace DS1103 for Electric Vehicle Power Consumption Modeling. In *21th Annual Conference Proceedings Technical Computing Prague 2013*. Prague: HUMUSOFT, 2013, p. 24. ISSN 2336-1662.ISBN 978-80-7080-863-4.