HUNGARIAN JOURNAL
OF INDUSTRIAL CHEMISTRY
VESZPRÉM
Vol. 39(1) pp. 57-62 (2011)

VEHICLE DYNAMICS BASED ABS ECU TESTING ON A REAL-TIME HIL SIMULATOR

K. ENISZ✉, P. TÓTH, D. FODOR, T. KULCSÁR

University of Pannonia, Institute of Mechanical Engineering, Faculty of Engineering
8200 Veszprém, Egyetem u. 10., HUNGARY
✉E-mail: krisztian.enisz@gmail.com

Satisfying the ever increasing needs for safety and environmental issues the electronic control units (ECU) of today vehicles become more and more complex [1, 2]. Regarding the controllability and safety requirements posed against automotive industry, the cost efficient ECU testing and the development of the software sensors becomes an important issue. The aim of this paper is to present the setup a HIL (hardware-in-the-loop) simulation environment which makes the functional testing and development of ECUs possible, and test the realized hardware and software environment with the integration of a General Motors Anti-lock Braking System ECU.

Keywords: HIL simulation environment, ABS, ECU

Introduction

The cost efficient ECU testing becomes an important issue. The development of new algorithms and new ECUs are necessary because the safety requirements are ever increasing. The testing of the new methods with real vehicle on a test track is very important but very expensive and it takes too much time. To help the proof of concepts developments it is necessary to implement a test and simulation environment which makes easier to create and validate new algorithm for different types of vehicles and ECUs.

The aim of the research was to create a software and hardware environment which is easily reconfigurable and its training time for new users is relatively short.

The functions of the system will be introduced with the hardware and software integration of ABS ECU.

The architecture of the simulation environment

The simulator has got modular architecture (*Fig. 1* and *Fig. 2*) which makes it easily reconfigurable. The first part is a standard high performance PC with Windows 7 operation system. It can be used as standalone system without the other part of the simulator and makes it possible to test new algorithm without any ECUs of the vehicle. The core of the system is a vehicle dynamics based simulation software, the Tesis veDyna 3.10.4. This software is based on Matlab/Simulink and the new algorithms can be implemented under Simulink environment, and capable to create new maneuvers, parameterize different vehicle models and the environment like road surface etc.

The interface for the real time hardware is the National Instruments VeriStand 2010. It can handle the models which were built by the veDyna, and can make the connection between input and outputs of the model and the hardware devices. The signal modification or override is one of the most important feature of the VeriStand. It can prepare the signals for the model or device inputs by the modification of the signals by simple mathematical algorithms or more complex LabView models.

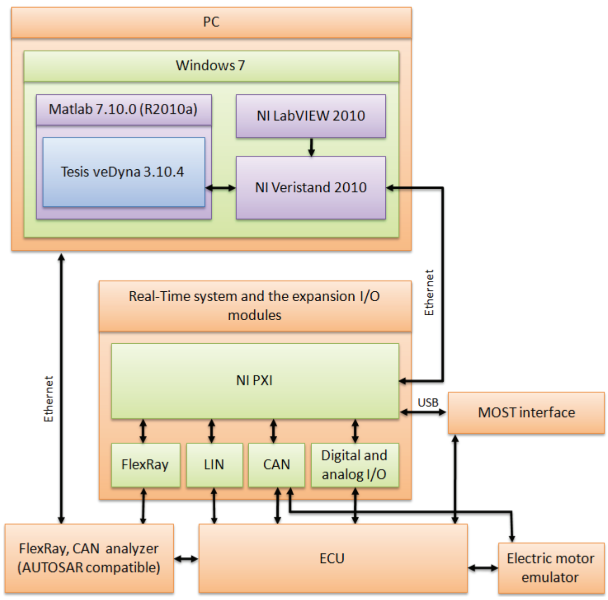

Figure 1: The anatomy of the HIL simulation environment

The second part is National Instruments based PXI real-time system. It has modular inputs and outputs and can be extended by National Instruments PXI and Compact RIO cards, like CAN, FlexRay, digital and analog IO cards. The PXI system has external interfaces like Vector MOST and TTech FlexRay interface which can be used without the PXI system, with a standard PC.

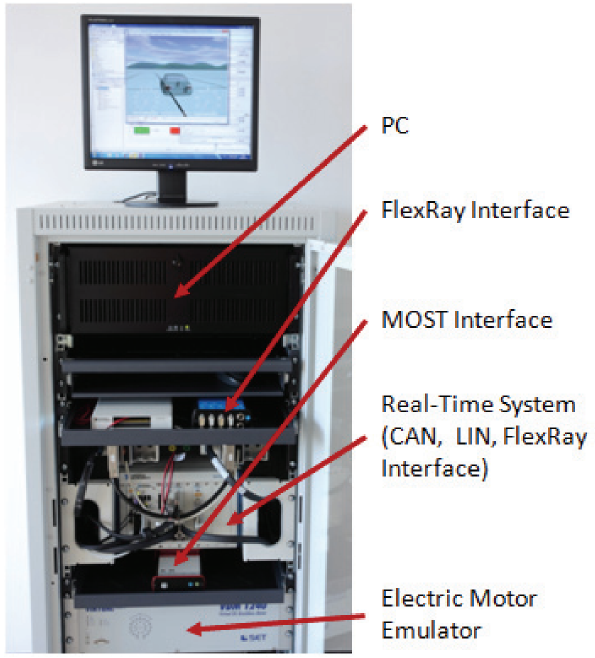

Figure 2: The physical appearance of the simulation environment

The third part is an electric motor emulator, which can emulate different types of electric motors used by electric cars driving systems.

The Integration of an ECU

After the realization of the simulation system, the very important step was the integration of an ECU *(Fig. 3)*.

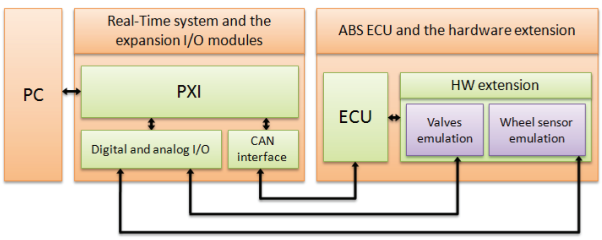

Figure 3: The hardware integration of the ABS ECU

The hardware integration

The ECU was extended with simple and small electric circuits to emulate the wheel speed sensors *(Fig. 4)*, the valves *(Fig. 5)* and the motor *(Fig. 6)*. The motor and the valves are controlled by the ECU, the Real-Time system only measured the states of these components. The wheel speed sensors are controlled by the simulator and a digital output switching the voltage on/off according to the provided wheel speed to generate the pulses with the correct frequency for the ABS.

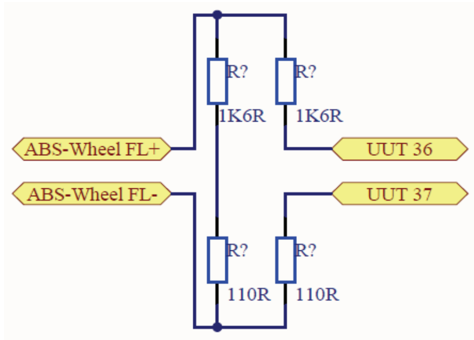

Figure 4: The emulation of the wheel speed sensors

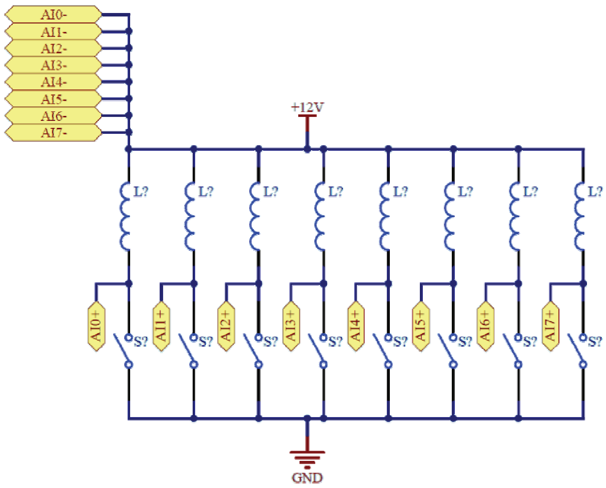

Figure 5: The emulation of the valves

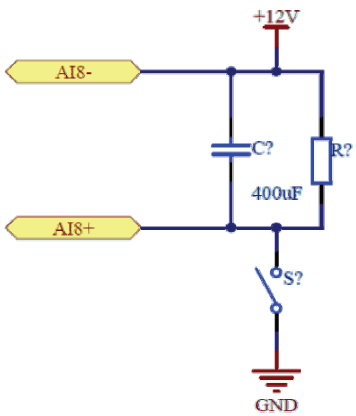

Figure 6: The emulation of the motor

The software integration into the Tesis veDyna

The brake system part of the original veDyna Simulink model was modified to handle the incoming valve states, and modify the brake pressure at the wheels according to the valves position *(Fig. 7)*, independently for each wheel.

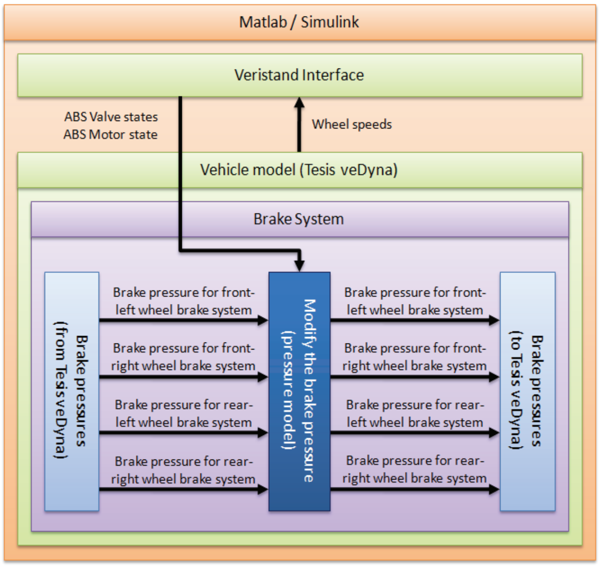

Figure 7: The software integration of the ABS ECU into the Tesis veDyna

The main equation of the pressure modification algorithm is (Eq. 1):

$$p(n) = p_{target}(n) - (p_{diff}(n))e^{\frac{-t(n)}{\tau}}, \qquad (1)$$

where:

n – The current simulation step

p(n) – The modified brake pressure for the wheel.

$p_{target}(n)$ – The target brake pressure

$p_{diff}(n)$ – The difference between the current and the target brake pressure

t(n) – The current time of the cycle

τ – The time constant

The algorithm has six different part as can be seen on the Matlab / Simulink chart *(Fig. 8)*:

- state selection *(Fig. 9)*
- main pressure change check *(Fig. 10)*
- cycle interval calculation *(Fig. 11)*
- target pressure calculation *(Fig. 12)*
- pressure difference calculation *(Fig. 13)*
- brake pressure calculation *(Fig. 14)*.

The state selection part sets the current state corresponding to the valves position and sign that the valves states is different from the previous valves state (Fig. 9).

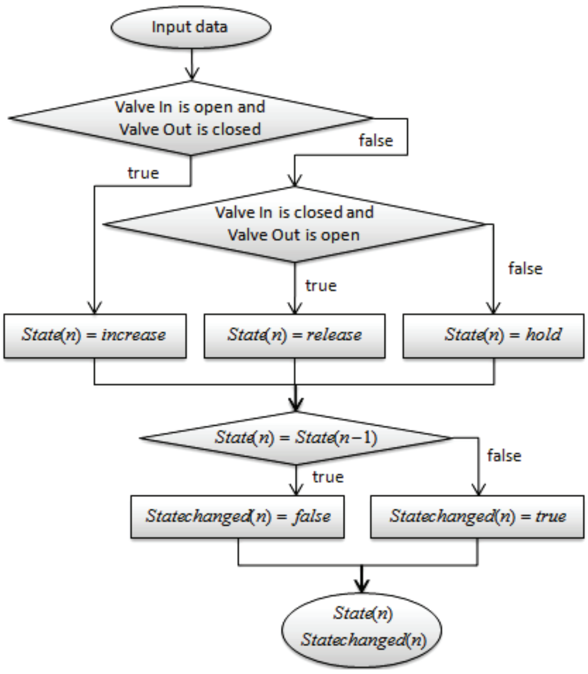

Figure 9: The calculation of the State(n) and State changed(n) parameters

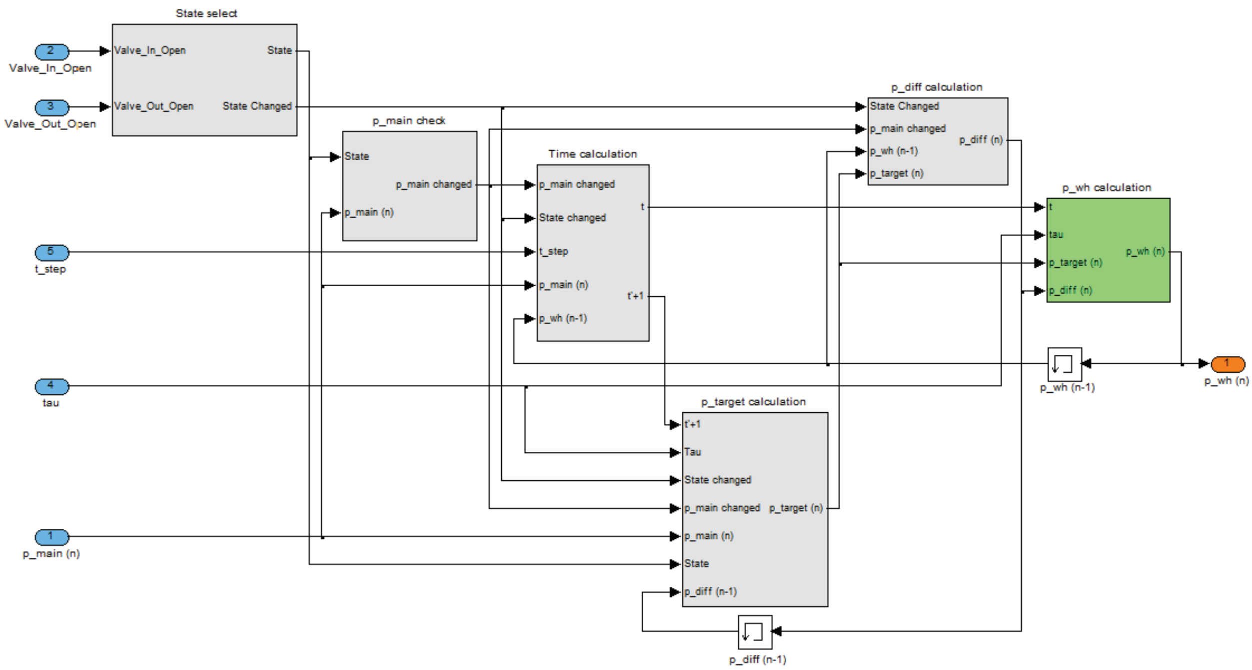

Figure 8: The pressure modification model for one wheel

The main pressure change check is necessary because if the driver increase or decrease the brake pressure the model cannot follow correctly this linear change. The problem was solved with a small modification to build up the linear pressure with small steps (where the step size is $p_{max\ change}$ and the p_{main} is the unmodified brake pressure for the wheel) (Fig. 10).

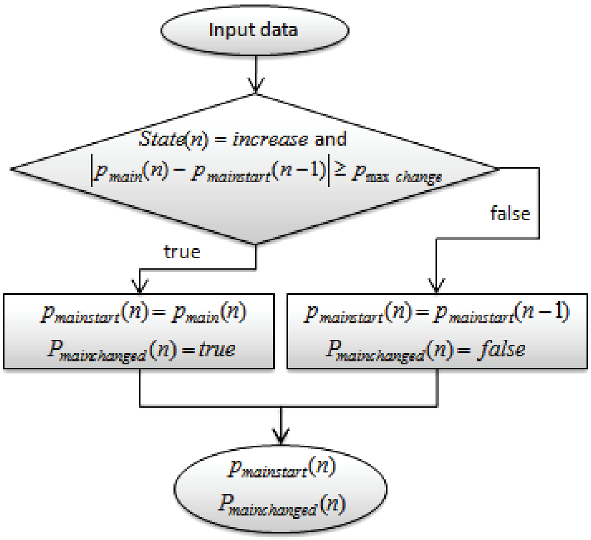

Figure 10: The calculation of the $p_{main\ start}(n)$ and $p_{main\ changed}(n)$ parameters

The cycle interval calculation module gives values of the current cycle interval, where the t_{step} is the simulation step length (Fig. 11).

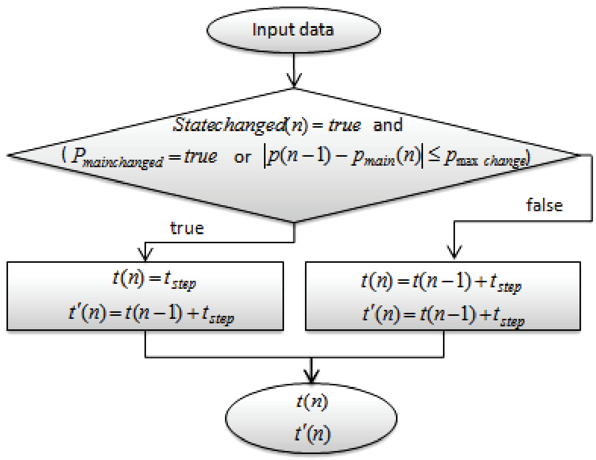

Figure 11: The calculation of the t(n) and t'(n) parameters

The target pressure calculation part selects the target pressure (Fig. 12).

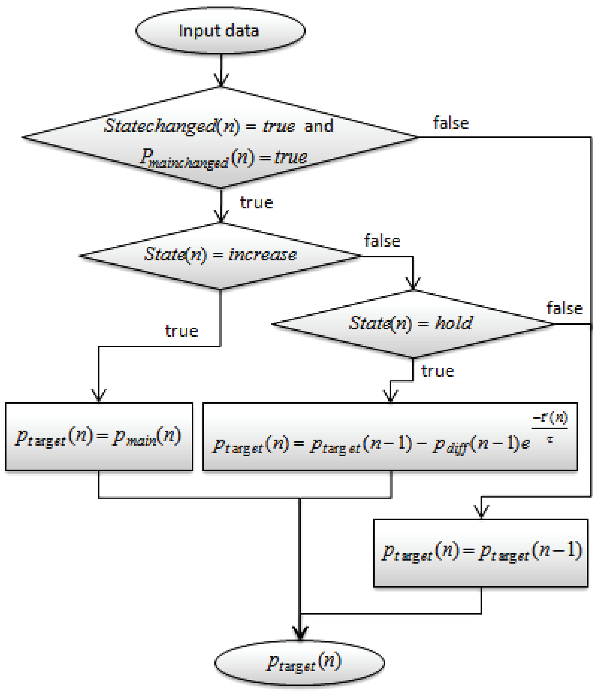

Figure 12: The calculation of the $p_{target}(n)$

The pressure difference calculation part calculates the difference between the current and the target brake pressure (Fig. 13).

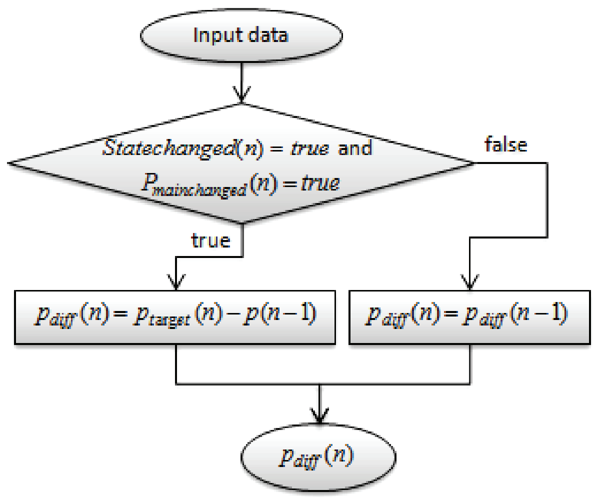

Figure 13: The calculation of the $p_{diff}(n)$

The pressure calculation part calculates the modified brake pressure (Fig. 14).

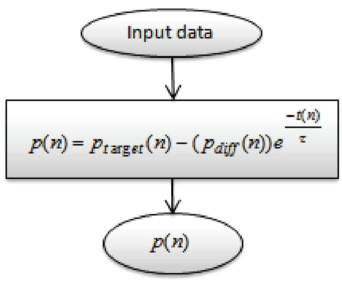

Figure 14: The calculation of the p(n)

After the modification of the veDyna model the next step was the creation of Veristand project. The compiled model is compatible with the Veristand so the input and output signals can be connected to the devices inputs and outputs. The conversion of wheel speed signals is necessary because the model provides the wheel speeds in km/h but the ABS ECU expect it frequency modulated pulses (Eq. 2) (switch on and off the voltage with a frequency).

$$f = \frac{\frac{v}{3.6}}{k} n_{teeth}, \qquad (2)$$

where:

v – The vehicle velocity

k – The circumference of the wheel

n_{teeth} – The numbers of the teeth of the speed measure disc

Another important setting is the emulation of the ECU starter CAN frames.

The final step was to create a Veristand user interface for the simulation to observe and change the parameters *(Fig. 15)*.

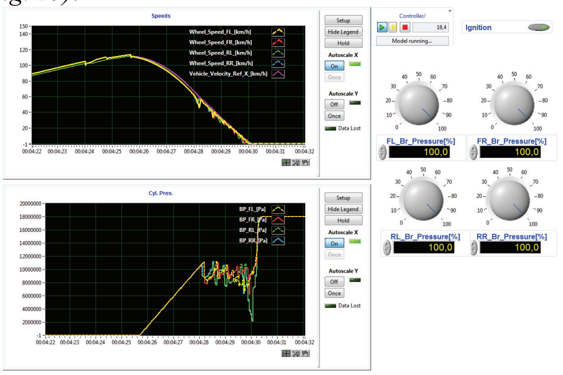

Figure 15: The speeds and pressure page of the user interface

Results

Based on the developments a real-time vehicle simulator was realized to create an integrated hardware in the loop (HIL) simulation environment in which new theoretical research results can be easily validated.

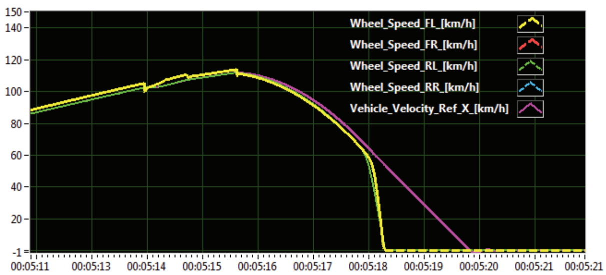

Figure 16: The vehicle velocity (km/h) and wheel speeds (km/h) without the ABS ECU (emergency braking from 115 km/h)

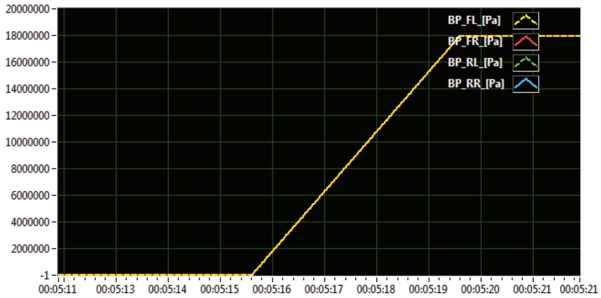

Figure 17: The brake pressures (Pa) without the ABS ECU (emergency braking from 115 km/h)

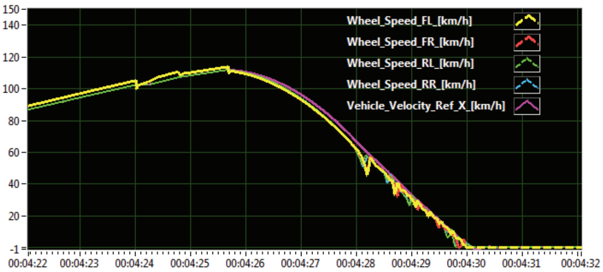

Figure 18: The vehicle velocity (km/h) and wheel speeds (km/h) with the ABS ECU (emergency braking from 115 km/h)

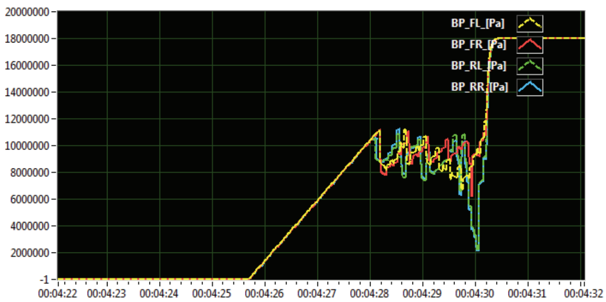

Figure 19: The brake pressures (Pa) with the ABS ECU (emergency braking from 115 km/h)

The behavior of the ABS ECU based test system was in concordance with expectations *(Fig. 16., Fig. 17., Fig. 18.* and *Fig. 19.)* but some small further development needed to refining the simulation. Another improvement should be on the pressure model taking into consideration of the motor state of the ABS.

ACKNOWLEDGMENT

We acknowledge the financial support of this work by the Hungarian State and the European Union under the TAMOP-4.2.1/B-09/1/KONV-2010-0003 project and the intensive technical support of Continental Teves Veszprem and National Instruments Budapest.

REFERENCES

1. P. E. Rieth, S. A. Drumm, M. Harnishfeger: Electronic Stability Program: The brake that steers, Verlag Moderne Industrie, 2002, 16–26
2. F. Gustafsson: Automotive Safety Systems, Replacing costly sensors with software algorithms, IEEE Signal Processing Magazine, 26(4), 2009, 32–47