Comfort analyses of the Hydractive suspension using a driving simulator

S. Barale[1], P. Vars[1], J. Guillet[2], M. Alirand[3], J. Lagnier[3], P. Toulemont[3]

(1) : Vélizy, PEUGEOT CITROEN SA
+33 1 57 59 31 03
stephane.barale@mpsa.com,
philippe.vars@mpsa.com

(2) : Paris, ALTEN
+33 1 57 59 89 95
jerome.guillet@ext-mpsa.com

(3) : Lyon, LMS Imagine
+33 4 77 23 60 30
marc.alirand@lmsintl.com
Julien.lagnier@lmsintl.com
'ierric.toulemont@lmsintl.com

Abstract – Peugeot Citroën SA is willing to build a virtual process based on model simplification techniques for component and control design using MiL, SiL and HiL. This process can be extended with the use of the SHERPA driving simulator for early phases of the process since a human in the loop is always a benefit when ride comfort is of concern. The modularity of the existing models seamlessly allows running the existing models in the driving simulator. Due to the complexity of the complete model, using the power of four CPUs was required but this did not affect the simulator performances. Demonstration on the static driving simulator was done on one hydraulic suspension architecture. However, work remains on model robustness which prevents right now running on the dynamic driving simulator.

Key words: ride comfort, cosimulation, control logic testing, hydraulic semi-active suspension.

1. Introduction

Peugeot Citroen SA (PCA) is well known for the comfort capability of its hydraulic suspensions and mostly the Hydractive one. As everyone knows, the Hydractive suspension is a two states semi active suspension including a self-leveling capability. An ECU controls the two states and corrects the height of the vehicle. PCA is thinking of a process in suspension design since many years [Ney1]. Recently a extended global MiL (Model in the Loop), SiL (Software in the Loop) and HiL (Hardware in the Loop) process to design the components of the hydraulic suspension and to design and validate the control logic has been explored [Bar1, Bar2]. However in very early phases of the design process, being able to test different types of control logic on a realistic model able to run real time on a driving simulator with a real driver could be of great interest.

Integrated design and engineering methods based on virtual testing are becoming standard practices in product and control design process. These methods support the development of mechatronic products and should address the challenges posed by their multi-disciplinarity and controller integration. Despite this "ideal" vision, often active functions are treated as add-ons potentially developed independently from the design of the system they are controlling. To integrate in a seamlessly approach system and control, PCA has elaborated a MiL/SiL/HiL process based on model simplification technique allowing first to virtually design the system to control, second to simplify the detailed models used in system design to integrate them within the controller design process targeting HiL testing. Variant analysis, performance optimization, control, component, subsystem and system level validation, and finally system integration must become intrinsic parts of a standard vehicle engineering process. This process coupling the simulations and tests is essential to reduce time to market. The challenge of this process is to enable a mechatronic system engineering approach that can be used throughout the complete design process, based on scalable and interoperable simulations. Interoperability requires common frameworks for development and exchanges: a multidisciplinary software platform sufficiently understandable and open with well-described interfaces to a control software.

Even if the process described in [Bar2] should target more generally mechatronic systems, the application to explore the concepts developed and used is the two states semi active Hydractive suspension including self-leveling capability. Comfort being the main interest of the suspension analyses, what could be the benefit of bringing the models on a driving simulator? First of all, analysing ride comfort on a driving simulator is not new [Hea1, Koh1]. At that time, mainly low frequency range was of interest with bouncing, pitch and roll as the main dynamic contributions and thus very low frequency range comfort (and handling) analyses were of concern. Nowadays, there exist specialised ride simulators with dedicated (Stewart or not) platforms [One1]. These simulators are able to go in high frequency domain (30-40 Hz in vertical direction). As mentioned in [Kad1], the application fields of driving simulators are human machine interactions, active safety research and vehicle dynamics experiments. In this last field, having an idea of the driver feeling in very early phases of the control process becomes of interest and even more when comfort is of interest. In [Mae1] driving on digitalized road looks to be of importance for comfort since as defined by the ISO 2631 and the NASA [Lea1], 4-8 Hz is the frequency band that affects the most ride comfort of human body in vertical direction. Regarding the performances of the driving simulator of concern, the band width in vertical direction is limited to about 10 Hz and the models are targeting 0-50 Hz range. The platform of concern is even lower frequency than the frequency range of the model content but both are in the range of interesting frequencies regarding the ISO and the NASA. Another interest for PCA is to introduce a human in the loop to explore the benefit at low frequency range where the two states switch affects the roll dynamics (low frequency steering wave inputs). Tuning up front the controller with a driver in the loop (DiL) could help gaining insights of the driver perception and should reduce system and its controller integration time.

After presenting some of the models developed for the MiL/SiL/HiL process and the model architecture used on the ds1006 HiL platform, the model architecture adaptation for the SHERPA driving simulator is introduced. The model is tested on the driving simulator for simple inputs, just to verify the capability to implement the models on 4 cores of the computer used by SHERPA. Discussion is than given on the opportunity to make further steps in using the driving simulator to further explore comfort with hydraulic suspensions and continuous semi active dampers.

2. Vehicle and hydraulic suspension model running in ds1006 HiL bench

For offline (design and MiL) and HiL testing, low frequency comfort analyses were the main target with a frequency range of 0-50 Hz. For this frequency range, the vehicle model includes the engine on its mount, the carbody torsion as well as the dynamics of the damper rod in the vertical direction [Bar1]. In order to fix the idea, the carbody torsion is around 15 Hz, almost similar to the engine bouncing mode in vertical (around 20 Hz) and the dynamics of the damper rod is about 40-50 Hz. Since the mechanical model matches the 0-50 Hz frequency range, the model of the hydraulic suspension should also be detailed to become sufficiently accurate in the same frequency range. The model of the hydraulic circuit of the suspensions (front and rear) and the way to simplify it has been explained in [Bar2]. Note that five suspension architectures were targeted for the analysis. The architecture in Fig 1 will be used in this paper to demonstrate how to implement and test the suspension architectures on the driving simulator. The architecture in Fig 1 is not the most complex but includes all the required elements to show the technique used: an electro pump assembly controlling the self-leveling at front and rear suspensions, two stiffness regulators controlling hard and soft for front and rear axles, piping systems and front and rear cylinders.

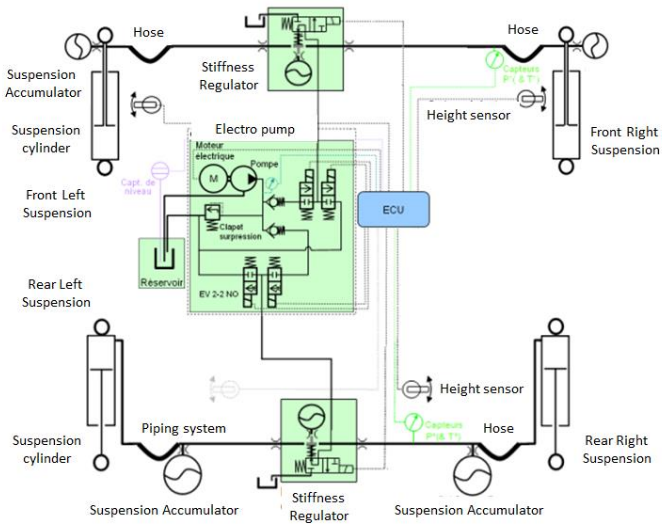


Fig. 1. Tested suspension architecture.

The complete model of the vehicle and the front and rear suspensions corresponding to the architecture in Fig 1 is shown in Fig 7 (in the Annexes). The modularity of the different constituents of the suspension has been analyzed in [Bar2]. This modularity allows building the five architectures in few clicks. As well since the complete model is clearly too complex to be run on one core, the model has been split in "modules" cosimulating between each other.

Going from MiL to SiL and then to HiL was possible thanks to model simplification tools [Bar2]. The complete model in Fig 7 was able to run real time on a ds1006 quadcore computer. The simulation architecture for MiL, SiL and HiL is shown Fig 2.

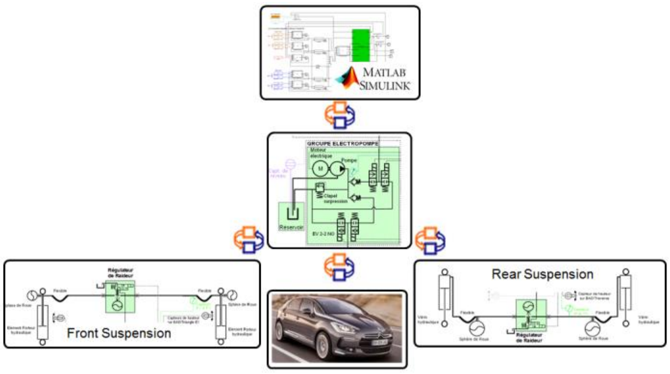


Fig. 2. Tested cosimulation architecture for HiL.

This simulation architecture corresponds to a cosimulation between Simulink for the control logic and AMESim for the electro pump assembly, Simulink being the master of the cosimulation. This choice was initially logical since the control logic is controlling the self-leveling using the electro-pump assembly (but also the two states semi-active suspension with the stiffness regulators). Note that for sake of CPU balancing, the "coil" models of the stiffness regulator (equivalent first order dynamics) were integrated in the electro pump model. Additionally to the Simulink - AMESim cosimulation, internal AMESim - AMESim cosimulations are also running, the electro pump assembly being the master of these cosimulations. The vehicle, the front and the rear suspensions are the slaves of the AMESim - AMESim cosimulations with the electro pump model. This architecture allows splitting and running the complete model (vehicle and suspensions) on the four cores of a ds1006 computer. The Simulink – AMESim exchange is done at 1 ms and corresponds to the sampling estimate of the future calculator (ECU). The exchange between the electro pump and the hydraulic suspensions is done at 0.1 ms required by the cosimulation stability. As explained in [Bar2], the exchanges between AMESim hydraulic models are done by pressure-flow rate variables around the electro-pump valves and the stiffness regulators. It was found that taking into account the pipe dynamics between these two components was of great help in order to stabilize the exchange rate to a value acceptable for real time application. Clearly the power exchange and the corresponding dynamics between the hydraulic models limit the cosimulation sampling rate. For the vehicle and the electro pump models, the sampling rate was 1 ms since the power couplings are mostly done via the wheel hop suspension mode, around 20 Hz. It is important to note here that when the damper rod dynamics is included (two different vehicle models were analyzed), the dynamics are changing due to a coupling between the mass of the damper rod and the hydraulic stiffness of the cylinder and piping stiffness. When the damping rod dynamics is taken into account, the sampling rate reduces to 0.1 ms

3. Model adaptation for the driving simulator

The internal process put in place tries to involve the driving simulator. Testing in early phases of the design process the suspension architectures and their related controller with a human in the loop was found of some interest. It is well known that the Stewart platform (or alike) used by driving simulators has a vertical direction bandwidth of action larger than in longitudinal and lateral directions. Even if the bandwidth of action in vertical of the SHERPA driving simulator is limited to 10 Hz, it was decided to test the rendering to analyze if this is sufficient or acceptable to "feel" the differences between a standard suspension, soft and hard in case of a switch on the Hydractive suspension.

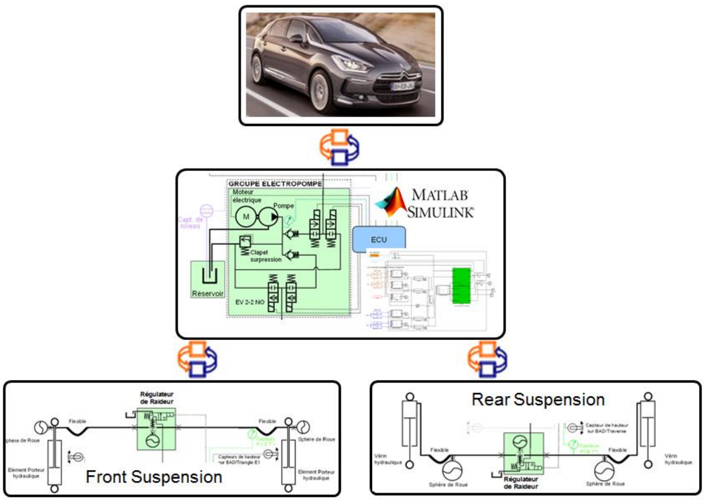


Fig. 3. Cosimulation architecture in SHERPA.

The model architecture used for the HiL bench is not really suitable for the driving simulator. The SHERPA driving simulator is using Simulink as a basis for generating the model running in the SCANeR environment [Okt1]. Thanks to Simulink, there was no adaptation to be done inside AMESim contrary to what has been done for [Fan1]. Note that thanks to [Fan1], directly interfacing AMESim to SCANeR is also feasible. Splitting the complete system in four models for off line testing (MiL and SiL) and for HiL in the ds1006 allows running each model on one CPU of the driving simulator. Thanks to the modularity of the complete model, the order of the Simulink – AMESim and AMESim – AMESim cosimulations was a bit changed to take into account the constraints of the driving simulator. The order of the cosimulations for the SHERPA driving simulator is now shown in Fig 3.

This time, the cosimulation has a master piloting a master-slave which pilots two slaves. All the models are in AMESim even the controller. Note that the controller initially in Simulink was integrated as an equivalent C-code for AMESim. The Simulink to AMESim interface was used to generate an encapsulated model of the controller thank to the C-code generation of Real-time Workshop. This technique was also used in [Fan1] to include the regenerative braking controller of the electric vehicle. A C s-function including the controller is also a possibility thanks to the usage of Simulink as main "interface". The two solutions are thus allowed by the SHERPA architecture.

The master of the cosimulation in Fig 3 is now the vehicle model. The master slave being the electro-pump assembly (and the controller), the front and rear suspension models are the two "ending" slaves. Previously (Fig 2), there was one AMESim master and three AMESim slaves. Now the architecture is a double stage AMESim cosimulation architecture: a master, a master-slave and two slaves. This was initially not planned (not expected) but this architecture works perfectly well. This simulation architecture was required as such since the SCANeR environment handle the inputs/outputs of the vehicle model to drive the moving platform.

4. Running within the driving simulator

The SHERPA driving simulator also called the dynamic driving simulator at PCA is shown in Fig 4. Normally only one CPU (over the 16) is reserved for the model. As shown in [Bar2], the model of Fig 7 is running on the four cores of the ds1006 used for HiL testing and one core was almost at its full load, the one corresponding to the front hydraulic suspension model.

Before testing on the dynamic driving simulator, tests have been done on the static driving simulator. This driving simulator has no moving platform. The platform corresponding to the vehicle is fixed to the ground. The environment is exactly the same as the dynamic driving simulator. This is generally the first test to do to verify first the real time capability of the model and second if the model is sufficiently robust to not put the dynamic driving simulator into troubles with numerical instabilities.

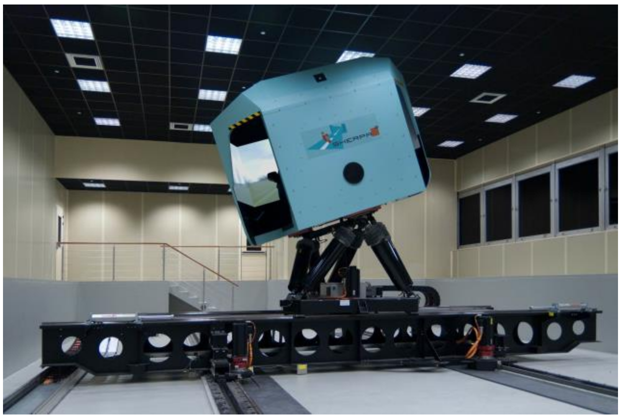


Fig. 4. The SHERPA driving simulator.

The static driving simulator is using a 8 cores computer. Running the model of Fig 7 gives the results shown in Fig 5. It is difficult to know which models and processes are running on which core since Windows is managing automatically the repartition. From Fig 5, it is clear that there is enough space to run all the models.

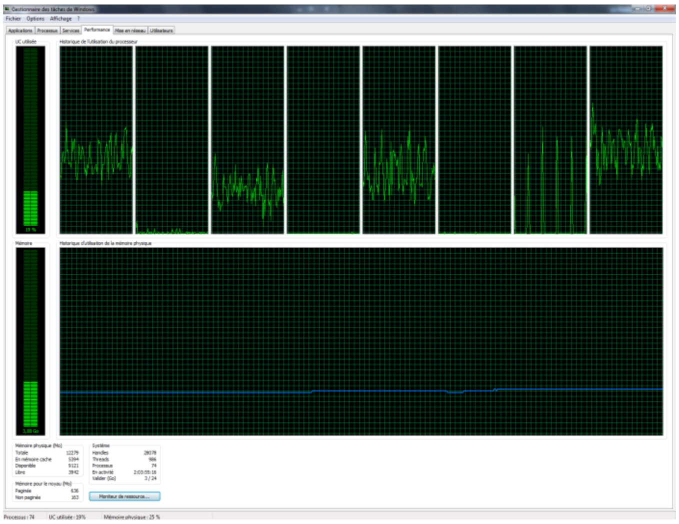


Fig. 5. Core loads of the static driving simulator.

The model of Fig 7 was tested on several roads and for different scenarios. Fig 6 shows an example of a standard test. In Fig 6 on top, the suspension control was piloted to hard and in Fig 6 at the bottom the suspension was piloted to soft. Soft can clearly be identified regarding the vehicle oscillations when driving on a "cleat". The introduction of the extra accumulator located inside the stiffness regulator as well as its additional damping valve allows the oscillations to be well damped. This is typically what the Hydractive two states suspension can provide as benefit for comfort.

It is now important to note that the cores of the static driving simulator are around 25 % more efficient than the cores of the dynamic driving simulator. From Fig 5, the core loads are less than 50 % meaning that running the models on the dynamic driving simulator will not be a problem. Remember that the core loads were around their maximum on the ds1006 [Bar2]. Despite this security margin, gathering two AMESim models to limit the complexity of the cosimulation architecture will result in reaching maximum core load even on the static driving simulator.

Fig. 6. Results from the static driving simulator.

Running the vehicle, the suspensions and the control logic inside the static driving simulator allows giving some insights on model capabilities and core loads. It also gives some information about model stability. Unfortunately, right now, the model looks too sensitive to be run on the SHERPA dynamic driving simulator.

5. Discussions

The static driving simulator was used first and some model instabilities have been encountered, typically when potholes, sharp cleats or trying to climb on sidewalks are of concern. Normal driving and the "comfort" road used in [Bar1] cause no difficulty. Up to now, it is not clear if these instabilities are coming from the tire model that does not filter enough the sharp road inputs or if it is the suspensions and electro pump hydraulic models that are numerically not enough robust. What is clear is that on the ds1006 computer, this kind of instability was never encountered. Despite this, it is necessary to explore in a deeper way the cosimulation and hydraulic model numerical stability to find the roots of the instability. Exploiting much more the HiL bench should give an idea of the robustness of the hydraulic models (and cosimulation architecture).

Coming back to ride comfort, analyses with a driver in the loop have not been done yet on the dynamic driving simulator. It is thus difficult to conclude on the real interest of having a driver "feeling" the effects of the control logic. However from the literature [Hea1, Kad1, Koh1], the effect at least for bounce, pitch and roll dynamics (low frequency range) should be obvious. The soft/hard switch should also be seen for Handling for instance with long wave length curves (steering wheel angle sine wave input around 0.1 to 0.5 Hz). Even if the self-leveling controller is not active like in [Ali1] and remains in a bandwidth of 10 to 15 seconds, testing the self leveling controller for long curves on highway for which the control logic can act against roll could be nice to explore. Again, the benefits should pop up in early phases of the control and component design process and thus prior to have the real components and final controller. Note that even if drivability and sine with dwell manoeuvers are difficult to reproduce in standard driving simulators, testing the control logic was partly the idea behind [Fan1] for electric vehicle and for ESC in [Fan2].

6. Conclusions

PCA is willing to put in place a virtual process for component design and control logic development and validation for suspension, a multi-functional system mock-up approach to build mechatronic systems. This process relies on MiL, SiL and HiL integration. Tentative for exploring the capabilities of different control laws in a driving simulator has been done to extend the virtual process to extra early phases.

At least, the results presented prove that running in the static (and dynamic) driving simulator a "complex" vehicle and Hydractive suspension model is possible. However, further testing is required to guaranty the usage of the model on the dynamic driving simulator. This step is essential to extend the virtual process (MiL, SiL, HiL) to Driver in the Loop providing solutions on both multi-physics simulation and control engineering integration levels.

7. References

[Ali1] Alirand M., Lachaize H., Lebrun M., "Study and analysis of an active self-leveling suspension". *1993 International Conference on Systems, Man and Cybernetics*, Le Touquet, France, October 1993, pp. 222-227.

[Bar1] Barale S., Plisson A., Guillet J., Lagnier J., Alirand M., "Improved functional modelling in comfort analyses for hydraulic suspension testing". *4th International Munich Chassis Symposium - Chassis.Tech Plus*, Munich, Germany, June 2013, pp. 1-22.

[Bar2] Barale S., Plisson A., Guillet J., Alirand M., Lagnier J., "Control logic testing for mechatronic systems: a process based on model simplification". *12th International Symposium on Advanced Vehicle Control*, Tokyo, Japan, September 2014, pp. 1-8.

[Fan1] Fang Z., Alirand M., Andre S., Denoual Th., Kemeny A., Reymond G., Jansson A., "Multi-objective analysis on a driving simulator applied to an electric vehicle: vehicle stability, handling and drivability". *SIA Congress on Vehicle Dynamics,* Mulhouse, France, October, 2011, pp. 1-10.

[Fan2] Fang Z., Kemeny A., Guo S.C., Deborne R., Denoual Th., Alirand M., "Electric vehicle's stability study on low friction road based on driving simulator". *Driving Simulation Conference,* Paris, France, September, 2012, pp. 1-9.

[Hea1] Heal S.F., Prasiloski J.C., "Ride characteristics of lightweight tracked vehicles", *Scientific Computer Division, report n°RRC-38*, February 1967, pp. 1-51.

[Kad1] Kading W., Zeeb E., "25 years driving simulator research for active safety". *10th International Symposium on Advanced Vehicle Control*, Loughborough, UK, August 2010, pp. 1-5.

[Koh1] Kohr R.H., "Real time automobile ride simulation". *Western joint IRE-AIEE-ACM Computer Conference*, New York, US, May 1960, pp. 285-300.

[Lea1] Leatherwood J.D., Barker L.M., "A user oriented and computerized model for estimating vehicle ride quality", *NASA Technical Paper n° 2299,* April 1984, pp. 1-41.

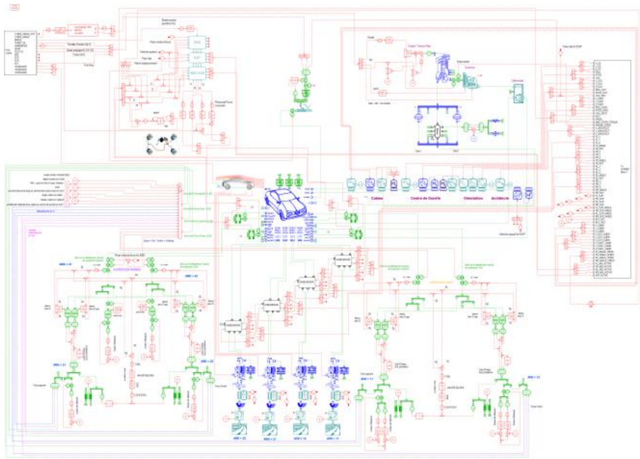
[Mae1] Maeda C., Kawamura A., Shirakawa T., Nakatsuji T., Kumada K., "Reproducibility of the vehicle vertical motion by KIT driving simulator using the actual measurement data". *Journal of the Eastern Asia Society for Transportation Studies, vol 6*, 2005, pp. 2734-2746.

[Ney1] Ney Y., Lebrun M., "Mechatronics in the automotive industry: Active suspension design using simulation", pp. 1-6.

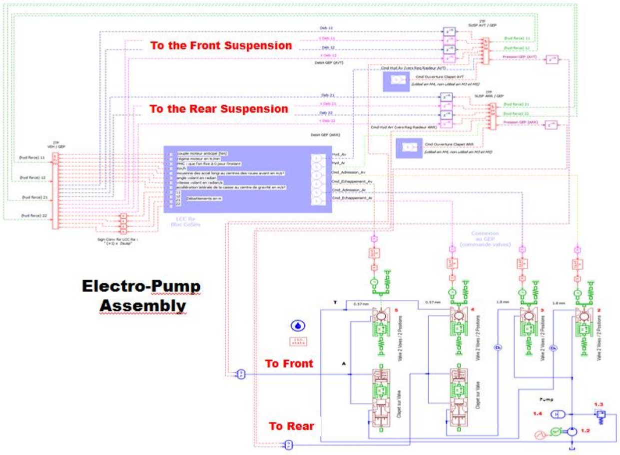
[Okt1] OKTAL, SCANeR is a trade mark of OKTAL, http://www.scanersimulation.com/.

[One1] Onesti P., "Verbindung von Human- und Hardware-in-the-Loop testing verkurzt entwicklungszeit". ATZ*, vol 111*, October, 2009, pp. 764-769.

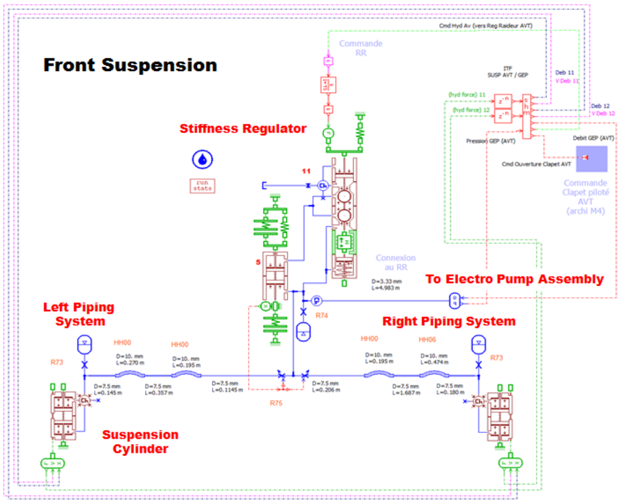
8. Annexes



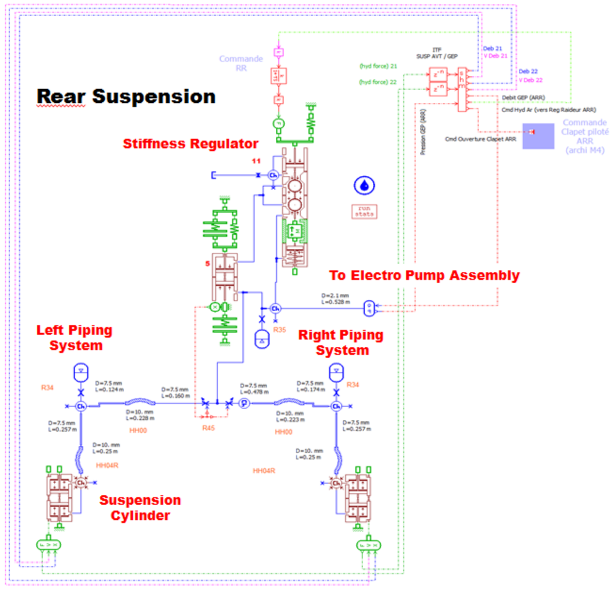
A – Vehicle (RK2 1 ms)



B – Electro pump assembly (RK2 0.1 ms)



C – Front suspension (RK2 1 ms)



D – Rear suspension (RK2 0.1 ms)

Fig. 7. The models running in cosimulation within the static driving simulator.