To my parents.

Abstract

Starting from vehicle targets and driver maneuvers, the AVL Requirement (RQ) Engineering approach is applied from vehicle level to component level in order to defined Electro and Electronic (E/E) requirements, functions and interfaces. From these, E/E components are defined and their sizing parameters estimated. After an evaluation phase, including communication between teams and checking commercial solution, next step is to specify component interfaces. The development process then, moves towards the signal and CAN-Matrix definition and, due to the passage from document-based approach to Model-Based System Engineering (MBSE) approach, also improvements on E/E modeling with SysML^{TM} are analyzed and applied. Finally, to complete and make the method more efficient, a parallel path leading to an integration process, is added and defined. For every methodology step, examples are provided to better understand how the step shall be applied and what the relationships between features/functions, architecture (and interfaces) and RQs are. Due to the necessity to apply the integration phase to the development process, a link between system engineering (system modeling) and simulation engineering (system simulation) shall be built up. From this last point, several considerations about simulation engineering, MBSE and the utilization of different tools are derived looking also to other possible future improvements that should be applied to reduce time and cost of the development process.

Contents

List of Figures

List of Tables

Vocabulary

Table 1: List of abbreviations and their definition

Abbreviation	**Definition**
ABS	Anti-lock Braking System
A/C	Air Conditioning
CAN	Controller Area Network
CU	Control Unit
Ctrl	Control
DCT	Dual (or Double) Clutch Transmission
DMF	Dual Mass Flywheel
E/E	Electro and Electronic
EMC	Electromagnetic Compatibility
EMS	Engine Management System
ESP	Electronic Stability Program
FC	Fuel Consumption
HMI	Human-Machine Interface
HV	High Voltage
HCU	Hybrid Control Unit
ICE	Internal Combustion Engine
LBC	Lithium-ion Battery Control (control unit)
LIN	Local Interconnect Network
LV	Low Voltage
LVBN	Low Voltage Board Net
MBSE	Model-Based System Engineering
MCU	Machine & Power- Electronic Control
NEDC	New European Driving Cycle
NVH	Noise, Vibrations and Harshness
PE	Power Electronics
PT	Powertrain
RLDC	Real Life Driving Cycle
RQ	Requirement

Table 1:continue on next page

Table 1:continue from previous page

Abbreviation	**Definition**
SOC	State of Charge (HV Battery)
SW	Software
SysML^{TM}	System Modeling Language
tbd	To be defined
TCU	Transmission Control Unit
WLTC	Worldwide harmonized Light duty driving Test Cycle

Glossary

Table 2: Terms and their definition from AVL PTE Glossary

Term	Definition
Component	Part of an element of the powertrain (further break-down of the elements of the powertrain). E.g. power electronics, Alternator, DCDC etc.
Control System	A control system is necessary to control the powertrain system or elements by functions. It consists of control unit(s), sensors and actuators and interfaces of the control system components; (busses, harness).
Control Unit	The "box"; A control unit consists of application software (control of physical system), basic software, electronic hardware (full scope of electronic hardware, including housing, connector). E.g. VCU, HCU, TCU etc.
Driving Maneuvers	Vehicle and Powertrain System Level: A driving maneuver describes a individual vehicle driving situation without changes of driving situation (quasi stationary vehicle, powertrain or component and function operation state) E.g. Constant Driving
Electromagnetic Compatibility	Electromagnetic compatibility (EMC) is the branch of electrical sciences which studies the unintentional generation, propagation and reception of electromagnetic energy with reference to the unwanted effects (electromagnetic interference, or EMI) that such energy may induce.
Element	Refers to the 5 core elements of the powertrain as specified by AVL (engine, transmission, e-machine, battery, controls)
Feature/ Sub-feature	Powertrain System Level: Definition of a specific interaction of vehicle and the powertrain system; related to vehicle, powertrain and component operation states. E.g. Boost = Electrical Acceleration Support.
Function	General: A function is an activity that transforms input to a desired output. A function is realized by hardware or Control (software) components; Control System Level: A software function to coordinate the elements and components to realize a feature on powertrain system level.

Table 2:continue on next page

Table 2:continue from previous page

Abbreviation	**Definition**
Phase	Further break down of the generations of the powertrain development process. Every generation includes the same phases (A-E): A: System Specification; B: Elements Specification; C: Element Development; D: System Integration, Verification & Validation; E: System Application.
Requirement	A statement identifying a capability, physical characteristic, or quality factor that bounds a product or process need for which a solution will be pursued [IEEE Std 1220-1994]
Requirement Engineering	To process of description and cascading from functional targets (attributes), features (sub-features) to the elements and components of the system. Requirements come from different sources such as OEM/customer, legal standards/regulations and the requirements engineering process itself (generation of requirements).
Use Cases	Vehicle and Powertrain System Level: A use case (UC) describes a complete driving cycle starting at vehicle standstill and stopping at vehicle standstill. It includes the interaction of driver, environment and vehicle. A UC consists of at least 2 sequences. It is always related to features, functions and/or functional targets. A UC can also be non-driving related. **Driver maneuvers are included in Vehicle Use Cases.**

Introduzione

Date le crescenti restrizioni normative riguardo ai consumi e alle emissioni, le case automobilistiche cercano sempre più di migliorare questi ultimi agendo sull'efficienza di motore e trasmissione e quindi sui software di controllo, sul consumo di potenza elettrica, sul peso del veicolo e sulla sua resistenza al moto. Allo stesso tempo, per essere competitive sul mercato, aggiungono applicazioni interattive, multimediali e di comfort per i passeggeri. Tenendo poi conto dei crescenti standard di sicurezza, altri componenti di sicurezza attiva vengono aggiunti al veicolo. L'insieme di questi ultimi e di quelli multimediali per il comfort, è chiaramente in contrasto con la riduzione del peso del veicolo e del suo consumo in termini di potenza elettrica. Per ottenere dunque una certa omologazione, e anche una maggiore attenzione da parte di una più ampia clientela (questo perché la gente si rende conto sempre più dei problemi attuali di inquinamento e allo stesso tempo il costi elevato del carburante porta ad un maggiore investimento iniziale per un veicolo con un limitato consumo), le case automobilistiche prediligono applicazioni ibride. Sotto quest'ottica, i componenti elettrici ed elettronici presenti nel veicolo (possono essere singoli o accoppiati tra loro e con altri meccanismi dando luogo a sistemi elettroidraulici o elettromeccanici) giocano un ruolo chiave sotto molti aspetti per raggiungere gli obiettivi prefissati quali per esempio prestazioni, consumi e guidabilità. Tali componenti infatti incidono sul consumo di potenza elettrica a bassa tensione e sul peso aggiunto al veicolo ma hanno anche importanti ruoli dal punto di vista della sicurezza e del controllo del veicolo e delle sue funzioni. Tuttavia è difficile determinare i requisiti di questi componenti, il loro dimensionamento e il loro ruolo sulle funzioni e il comportamento del sistema integrale, partendo dalle caratteristiche del veicolo richieste dal costruttore. Un altro punto fondamentale nella scelta dei componenti è la determinazione delle loro funzioni e caratteristiche a partire dalle manovre di guida che il veicolo deve sostenere. Queste vengono solitamente richieste dal costruttore e dipendono inoltre dal tipo di veicolo, infatti un veicolo 4X4 dovrà essere in grado di eseguire manovre/comportamenti diversi da una vettura utilitaria. Le principali manovre di guida del veicolo e le associate funzioni che il powertrain o altri sistemi del veicolo devono svolgere sono rappresentate in Figura 1 in funzione della velocità del veicolo.

Si può notare la differenza delle funzioni a seconda che il veicolo sia convenzionale (cioè con il solo motore a combustione interna) o ibrido elettrico (motore a combustione interna e motore elettrico). Tali funzioni sono riferite ad un veicolo Full-Hybrid, cioè un veicolo in cui il motore elettrico da solo è in grado di far avanzare il veicolo fino una certa velocità e/o per una certa distanza data una velocità costante minore della massima possibile in solo modalità elettrica. Le funzioni aggiuntive di una vettura ibrida quali Start & Stop, Boost elettrico, frenata rigenerativa e guida solo elettrica permettono di ridurre il consumo di carburante e allo stesso tempo di massimizzare l'efficienza dei componenti. Tuttavia comportano anche un'elevata complessità nei software di controllo e l'interazione dei singoli componenti diventa essenziale per il corretto funzionamento del sistema nonché la corretta esecuzione di ogni funzione. Pertanto, nuovamente si deriva la necessità di dimensionare correttamente i componenti base del veicolo affinché il sistema completo svolga le funzioni assegnate e raggiunga gli obiettivi prefissati, in termini di performance, guidabilità, sicurezza, consumi ed emissioni. Per migliorare il metodo AVL, quest'ultimo è stato applicato a diversi progetti, tra cui un veicolo Full-Hybdid con configurazione denominata P2. Tale configurazione del powertrain, rappresentata schematicamente in Figura 2, consiste nell'inserimento della macchina elettrica tra il motore a combustione interna e la trasmissione. Quindi, un elemento di separazione (solitamente una frizione) è necessario per disaccoppiare macchina elettrica e motore garantendo così al veicolo di guidare solo elettrico.

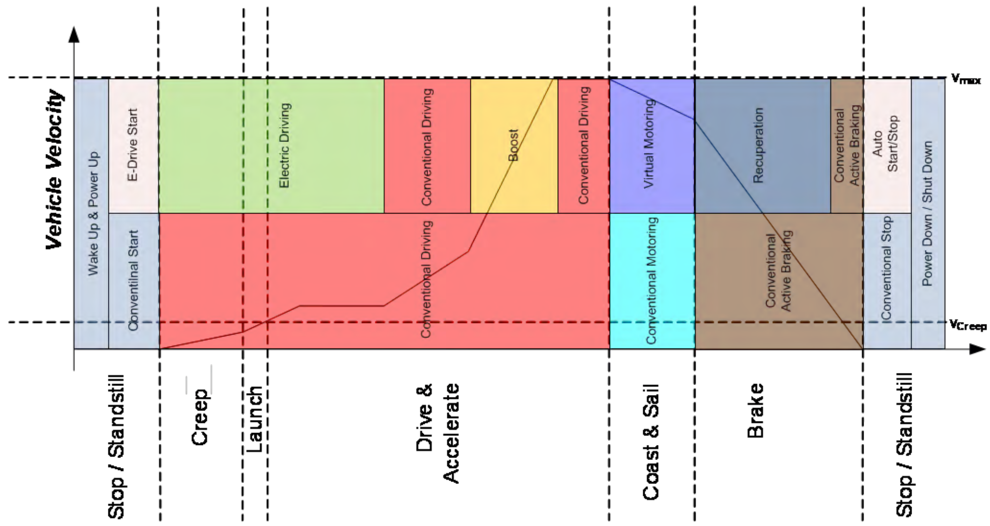

Figura 1: Manovre di guida e relative funzioni del veicolo

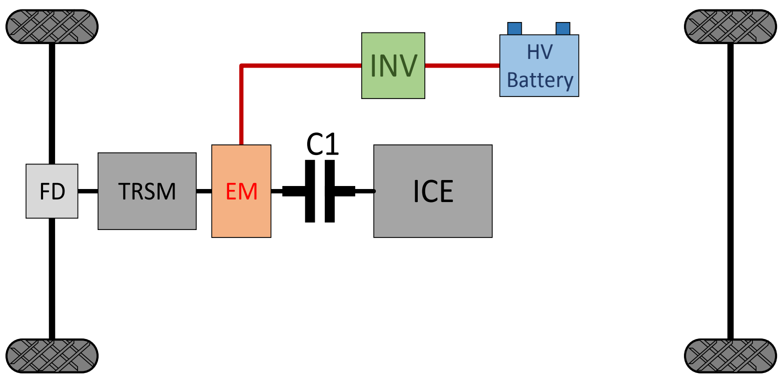

Figura 2: Configurazione P2 del powertrain ibrido

La configurazione P2 è comunemente utilizzata per applicazioni Full-Hybrid in veicoli già esistenti e nonostante i pesanti vincoli nel dimensionamento in termini di dimensioni geometriche di macchina elettrica e elemento di separazione, è preferita per i minori costi nella produzione di serie non che l'adattabilità a differenti tipologie di motori. Ulteriori informazioni sul veicolo sono le seguenti: la trasmissione è a doppia frizione e al motore è stato applicato il ciclo Atkinson, il quale permette di ridurre il consumo specifico a fronte di una diminuzione della potenza massima, la quale è però bilanciata dalla potenza aggiuntiva del motore elettrico. Partendo dal livello 0, il livello del veicolo, requisiti e funzioni vengono derivate fino al livello 3, livello dei componenti. Nel caso specifico qui riportato, partendo dalle funzioni del veicolo ed i suoi requisiti, si scendono i vari livelli fino a determinare requisiti e funzioni di alcuni componenti passando poi al loro dimensionamento e specificazione.

I componenti considerati sono:

- pompa del vuoto del sistema frenante convenzionale;
- la pompa elettrica dell'olio;
- i solenoidi;
- i sensori di pressione.

Gli ultimi tre componenti appartengono al circuito idraulico di controllo e lubrificazione della frizione del modulo P2 (cioè la frizione che accoppia e disaccoppia motore a combustione interna e macchina elettrica). Dopo una breve definizione dei concetti di System engineering e Model-Based System Engineering (in particolare gli aspetti positivi dell'applicazione dell'approccio MBSE), un programma utilizzato nel system engineering e in questa tesi, Artisan Studio SysML^{TM}, è introdotto con le sue principali tipologie di diagrammi ed il loro utilizzo/significato nella modellazione del sistema (in questo caso il veicolo). Quindi, il metodo di AVL di derivazione dei requisiti di sistemi, sottosistemi e componenti del veicolo a partire dalle richieste del costruttore, da vincoli legislativi su emissioni e sicurezza, dall'ambiente di utilizzo del veicolo nonché dagli obiettivi finali di consumo e performance, viene prima descritto in maniera generale e poi applicato a scopo d'esempio ai componenti elettrici, elettronici sopracitati per meglio comprendere come avviene l'assegnazione di funzioni e requisiti ad un sistema o un componente. Inoltre la sua applicazione consente di individuare eventuali punti deboli di tale metodo ma anche possibilità di sviluppi futuri. Il metodo AVL è inoltre integrato con delle fasi che riguardano la specificazione dei componenti elettrici ed elettronici e dei loro collegamenti, siano questi tramite reti CAN, FlexRay, LIN, ecc. o tramite semplici cavi elettrici di potenza e segnale. Una volta completata la specificazione dei sistemi, sottosistemi e componenti, una fase veramente importante per il processo di sviluppo del powertrain è quella rappresentata dal processo di integrazione. Tale processo consiste in una serie di simulazioni e prove, sia di tipo software che fisiche in laboratorio, banco prova o circuito di prova, per convalidare i componenti scelti, cioè verificare che grazie ad essi i vari sistemi prima ed il veicolo poi soddisfino i requisiti. Da qui nasce l'importanza di collegare il modello del sistema, che comprende la struttura del sistema, i suoi componenti e i relativi collegamenti, le loro funzioni e altre informazioni quale mappe, grafici e dati caratteristici, con le simulazioni di singoli componenti e sistemi. Ovviamente queste simulazioni, se fatte a livello di software, vengono effettuate con differenti programmi. Ne deriva dunque la necessità di creare collegamenti (cioè interfacce a livello software) tra di essi e permettere così un elevato scambio di dati diminuendo da una parte il tempo impiegato nella costruzione del modello da simulare e dall'altra il numero di documenti creati e quindi riducendo le possibilità di errore nella scrittura e nella lettura dei dati stessi. Tutto questo in perfetto allineamento con l'approccio MBSE. Con riferimento alle fasi aggiuntive sulla specificazione del collegamento dei componenti elettrici ed elettronici e la loro modellazione usando SysML, durante lo svolgimento del lavoro di tesi un possibile metodo è stato sviluppato in collaborazione con degli ingegneri di sistema. Tale metodo da la possibilità di includere tutti i dati riguardanti i componenti elettrici ed elettronici nel modello del veicolo e permette anche la loro pubblicazione come elenco/specifiche. Gli studi fatti in proposito mirano a diminuire i documenti rilasciati e ad includere più informazioni possibili nel modello in SysML, sempre in accordo con l'approccio MBSE. Sempre dall'esperienza acquisita durante la modellazione con SysML, ne derivano ulteriori miglioramenti e sviluppi futuri che saranno poi elencati. Dall'applicazione della metodologia e durante lo svolgimento della tesi, diverse problematiche relative al metodo AVL sono emerse: queste sono principalmente legate da una parte alla definizione del sistema e dei suoi componenti e dei livelli in cui essi sono posti e dall'altra alla definizione delle funzioni che i vari sistemi devono svolgere. Infatti, tanto più queste sono dettagliate e tanto più semplice e veloce risulta poi l'applicazione del metodo e quindi la specificazione dei componenti con tutti i vantaggi che ne derivano in termini di tempo e quindi costi nel processo di sviluppo di powertrain. Da considerazioni su quanto appena detto, ne derivano ulteriori conclusioni oltre a quelle riguardanti metodi di modellazione dei componenti elettrici ed elettronici e loro architetture, l'applicazione totale dell'approccio MBSE e l'importanza della comunicazione tra membri dello stesso dipartimento e membri di dipartimenti diversi. Infine, possibili sviluppi futuri riguardo la modellazione dei componenti E/E con SysML, la generazione di specifiche di questi partendo dal modello e l'importanza di collegare l'ambiente di modellazione con quello di simulazione, vengono analizzati e discussi.

1 Introduction

Due to the increasing powertrain (PT) electrification in hybrid vehicles, Electro- and Electronic (E/E) components play a major role in vehicle targets achievement. In fact, electric power consumption has a high effect on fuel economy and, at the same time, the increasing weight, due to components adding, badly affects fuel economy and vehicle performance. Therefore, it is critical to derive E/E component requirements from clearly defined vehicle and powertrain targets, to understand how those E/E components deliver hybrid functionalities and to balance requirements on all levels before components are approved and control strategies are finalized. Scope of this document is to guide system engineers into E/E components specification starting from customer wishes and vehicle targets, and simulation engineers to E/E integration and simulation. Furthermore, it is also desired to underline the importance of linkages between system engineering and system simulation and of the Model-Based System Engineering (MBSE) approach in order to achieve a transparent information flow and avoid sharing contradictory data. This paper describes a methodology, based on the current AVL requirement engineering approach and on MBSE, which allows the specification of E/E component. The method has been applied to development projects to confirm requirements and E/E specifications. These reference the powertrain development of a full hybrid vehicle with P2 powertrain configuration. This configuration, shown in Figure 1.1, includes a separation element and an electrical machine located between the Internal Combustion Engine (ICE) and the transmission. As a full hybrid powertrain, the electrical machine is able to propel the vehicle alone for a certain km range and up to a determined vehicle speed. Examples are provided in the paper to demonstrate the linkages and how the electrical components are specified and then simulated.

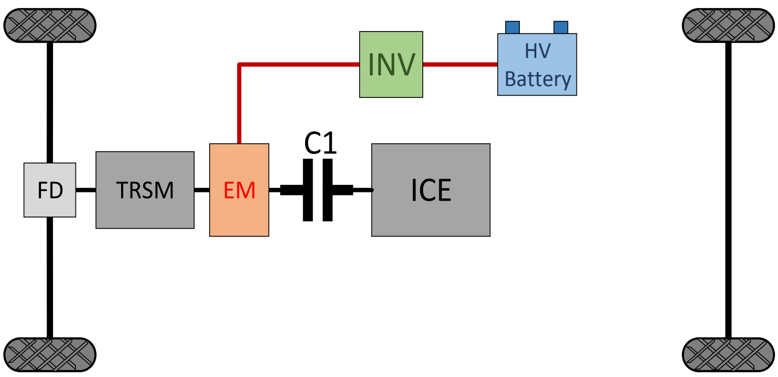

Figure 1.1: P2 powertrain configuration

1.1 Vehicle targets and driver maneuvers

Vehicle targets are the starting point for every development project, and from these requirements are formulated in order to achieve them. Vehicle targets usually given in PT development are fuel economy, durability, drivability, performance, safety, noise vibration and harshness (NVH), etc. At the same time, they are linked to some vehicle attributes like weight, driving resistances, fuel consumption, vehicle architecture, elements type, etc. In fact, if a conventional vehicle is hybridized (i.e. PT is electrified) and pure electric drive is performed to achieve a better fuel economy, this means that the weight increase for the component added and with the weight also the driving resistances increase and these affect vehicle fuel consumption and performance. Note that some targets are at the same time attributes (i.e. fuel consumption) and not all vehicle attributes are considered (e.g. vehicle color has no influence on powertrain development). Therefore, vehicle attributes as the weight and other important targets like cost, are also cascaded through system levels. Driver maneuvers are also important inputs for requirements cascading. In fact, they allow deriving vehicle features (e.g. electric driving) which then will be allocated to vehicle systems like powertrain, body and chassis and will be used to specify vehicle architecture and system interfaces. Driver maneuvers and the related vehicle features are shown in Figure 1.2. In the same picture, vehicle features are divided in conventional and hybrid features.

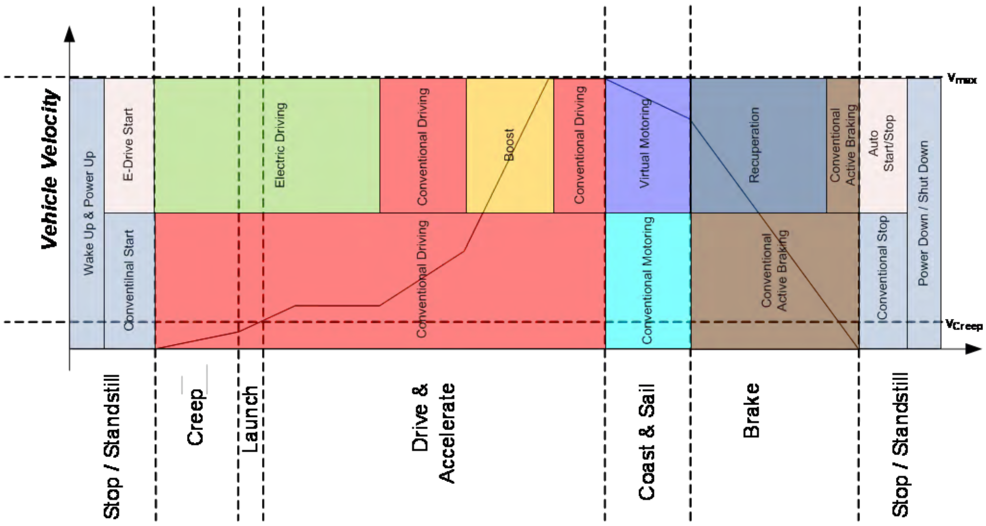

Figure 1.2: Driver maneuvers and related vehicle features

Before start explaining the methodology, a brief introduction to system engineering and system modeling is done in the next chapter. In particular way, the system modeling tool SysML is introduced.

2 SysML^{TM}

2.1 Model, systems engineering and MBSE Approach

In every development process is necessary to model the system/object that we want to develop in agreement with the stakeholder needs, the physical law, the technological possibilities, etc. A model is an abstract, or sometimes also concrete (prototype), representation of one or more objects that can be physically realized. It generally described a domain of interest. A key feature of a model is that it has not to contain all the details but only those are necessary for the purpose of the model. In every case, the model can be simple or more complex (more detailed model), but *"Essentially, all models are wrong, but some are useful."* [G. E. P. Box, 1987]. In fact is difficult that a model matches the reality. Furthermore a very detailed model is not always useful because of the high complexity. In general, a system consist of a set of elements that interact together, and can be viewed as a whole that interacts with its external environment. The purpose for modeling a system for a particular project must be clearly defined as the starting point of system engineering work, in terms of the expected results of the modeling effort, the stakeholders needs, the depth and fidelity of the model, etc. This scope should be achieve keep in account of the available knowledge, budget, tools and other resources. Systems engineering is a multidisciplinary approach used to model a system in order to achieve the stakeholder needs with less effort and cost as possible and representing a system that can be understand from the other persons involved in the development process.

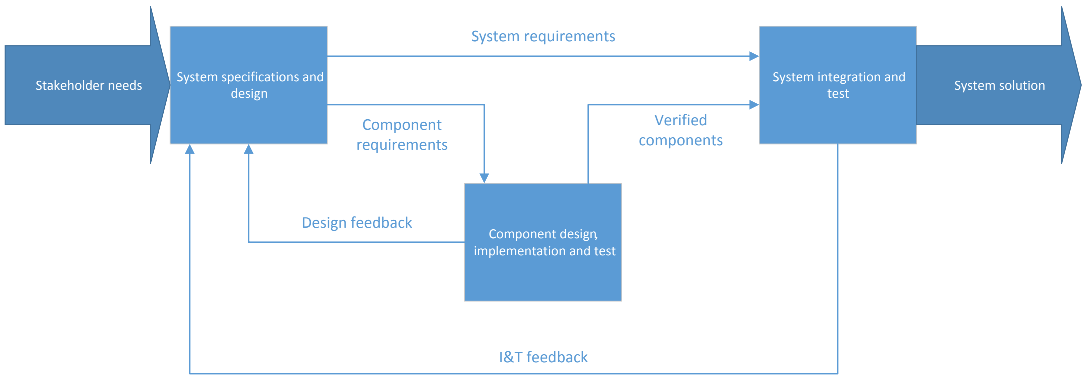

Figure 2.1: System Engineering technical process

Systems engineer, for starting to model a defined system as we can see in Figure 2.1 , have to know and define:

- system requirements (directly from stakeholders necessity);
- system environment and actors and their interactions with the system;
- system characteristics and structure;
- system behavior and main features;
- components requirements, interfaces, features and parameters (derived from the whole system).

The following steps are: validate components requirements and design with simulations, then integrate the components in order to simulate and validate system requirements. Systems engineering methodology is transitioning from a document-based approach, based on collect all the informations of a certain system, its requirements, parameters etc. in the document like paper, to a model-based approach like many other engineering disciplines have already done. In the second approach, all the informations available are integrate in the model of the system.

From International Council of Systems Engineering (INCOSE) *INCOSE*, the definition of MBSE: *"Model-based systems engineering (MBSE) is the formalized application of modeling to support system requirements, design, analysis, verification and validation activities beginning in the conceptual design phase and continuing throughout development and later life cycle phases. MBSE is a part of a long/term trend toward model-centric approaches adopted by other engineering disciplines, including mechanical, electrical and software. In particular, MBSE is expected to replace the document-centric approach [...]"*.

2.2 SysML^{TM}: definition, purpose and structure

The Unified Modeling Language^{TM} (UML^{TM}), introduced in the 1990's, is the modeling language used by software engineering to model software systems. The International Council of Systems Engineering (INCOSE) and the Object Management Group (OMG) proposed in 2001 an adapted version of UML for systems engineering. The production of SysML specification in 2006 take the interest of systems engineering community. The Object Management Group's OMG SysML^{TM} is based on UML and in Figure 2.2 we can see the interrelationship between SysML^{TM} and UML^{TM}: from one side not all the UML's content is used by SysML and from the other side Sysml adopt new tools that UML does not contain.

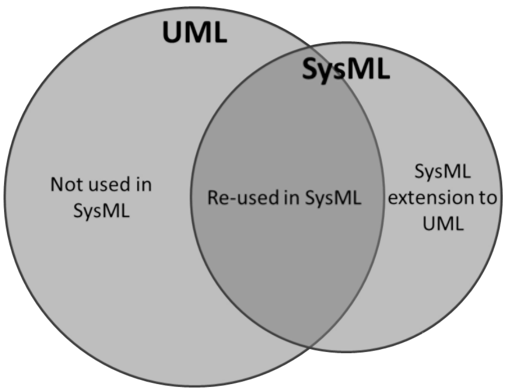

Figure 2.2: SysML^{TM} and UML^{TM} interrelationship from Atego, *SysML Tutorial v8-0-0.pdf*

OMG SysML^{TM} is a general-purpose graphical modeling language for specifying, analyzing, designing, and verifying complex systems that not only include hardware and software, but also data, people, and natural objects. In particular, the language provides graphical representations of the main aspects of the system: behavior, structure and parametrics. Furthermore SysML^{TM} support the practice of MBSE that is used to develop system solution in response to complex and often technologically challenging problems.

Two important things that we have to keep on mind are:

- SysML is a semi-formalized, graphical modeling language with specific semantics (meaning) and notation (graphical representation);
- SysML is not/does not contain a methodology or tool (is methodology and tool independent).

SysML^{TM} can represent the following aspects of systems, subsystems, components, etc.:

- structural composition, interconnections and classifications (hierarchy structure);
- function and behavior;
- constraints and performance properties;
- allocations between behavior, structure and constraints;
- requirements and their relationship to other requirements, design elements and test cases.

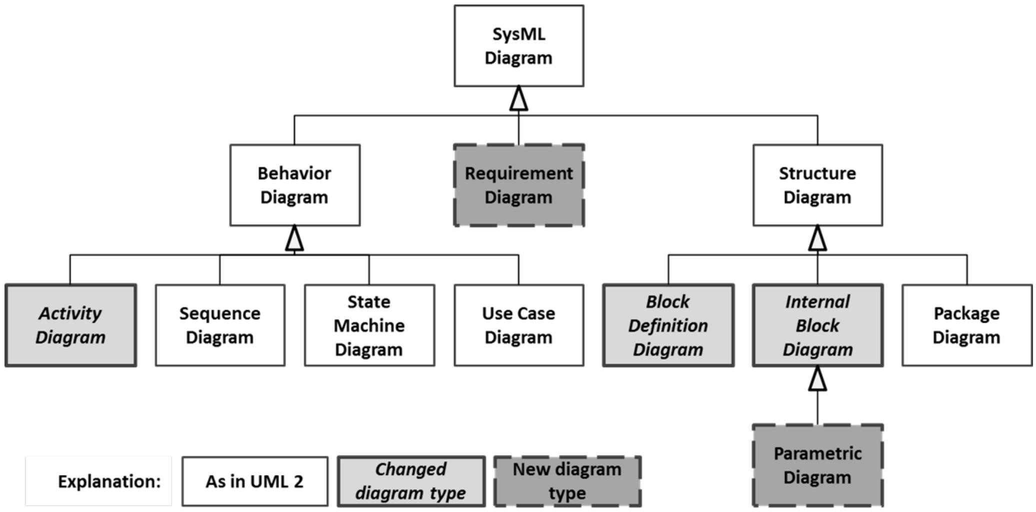

Figure 2.3: SysML^{TM} Diagram Taxonomy from Atego, *SysML Tutorial v8-0-0.pdf*

In SysML^{TM} are available nine diagrams as shown in Figure 2.3. A diagram graphically represents a particular aspect of the system model. Each diagram type and is description are summarize in Table 2.1.

The main parts of a system model are: structure, behavior, requirements and parametrics and after their creation, they have to be linked together. One way to do this is use requirements, but other ways are available in SysML^{TM}. So that, the main diagrams used in system engineering process are use case diagram, activity diagram, definition block diagram, internal block diagram, parametric diagram and then requirements. Use cases represent the functional system context and the top-level system behavior and are focused on specific working tasks in context. They are often used in a sense of business processes or work-flows and specify the system behavior from the view of an external actor (human-being, external system,...) A well-structured use case specifies a single, atomic behavior of the system, visualizes common behavior to corresponding use cases with the include relationship and visualizes exception with extend. In Figure 2.4 is shown an example of Use Case diagram with the actors (Driver and Maintainer), the use cases (Drive the vehicle, maintenance and repair) and the relations. In this diagram is possible add requirements to the use cases.

The activities of the system (i.e. the functions of the system) are modeled with two diagrams: the block definition diagram is used for building the system's functional breakdown structure (Figure 2.5) and contains the activities and the relations between them, and the activity diagram (Figure 2.6) that is similar to flow chart representations and contains start and end points, flows, decision nodes, merge nodes, joint nodes and activities.

Blocks, the basic element of structure in SysML^{TM} for entities and objects, e.g. system components, are used to build the structure of the system and subsystems and in this case are used both block definition diagram (Figure 2.7) and internal block diagram (Figure 2.8). The first one represents the system's breakdown structure and includes hierarchy e dependency representations, the second is used for describing the internal structure of a block and their interconnection with parts e ports (flow ports, standard ports,...). For blocks we can add several properties that define the hierarchical structure (part properties/parts) the dependencies

Table 2.1: SysML^{TM} diagrams and their description

Diagram type	**Description**
Package Diagram	represents the structure of the system, organized in package that contain the model elements
Requirement Diagram	represents system, functions and components requirements and their relationship to other requirements, design elements and test cases. Furthermore it supports requirements traceability
Activity Diagram (*ad*)	represents the behavior of the system and the order of the execution of the actions based on inputs, outputs and controls
Sequence Diagram	represents the behavior in terms of a sequence of messages exchanged between systems or parts of systems
State Machine Diagram	represents the behavior of an entity in terms of its transitions between states triggered by events
Use Case Diagram	represents functionality regard how actors use the system in order to achieve a set of targets
Block Definition Diagram (*bdd*)	represents a hierarchical structure with elements called blocks, and their composition and classification
Internal Block Diagram (*ibd*)	is similar bdd but represents interfaces and interconnections between the parts of a block
Parametric Diagram (*par*)	represents constraints on property values, such a formulas, and is used to support engineering analysis

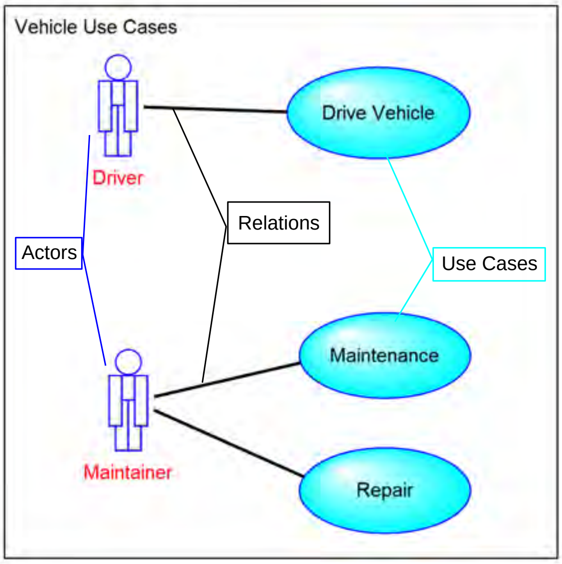

Figure 2.4: Use Case Diagram

between blocks, and the quantifiable characteristics of a block (e.g. velocity, weight, dimensions etc.).

Parametric diagrams are used to add mathematical relationships and constraints to the model, in particular between parts and value properties of blocks. Constraint blocks, that are typically reusable specifications in the model, define determined constraints, like formulas,

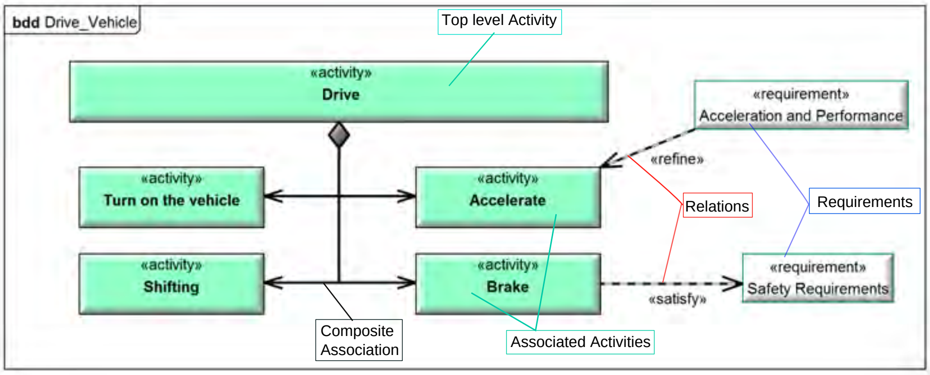

Figure 2.5: Block Definition Diagram for Activities

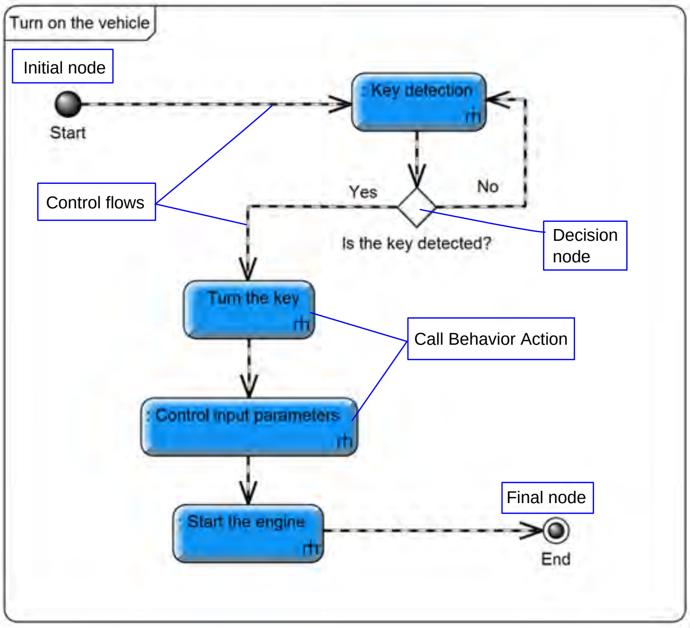

Figure 2.6: Activity Diagram

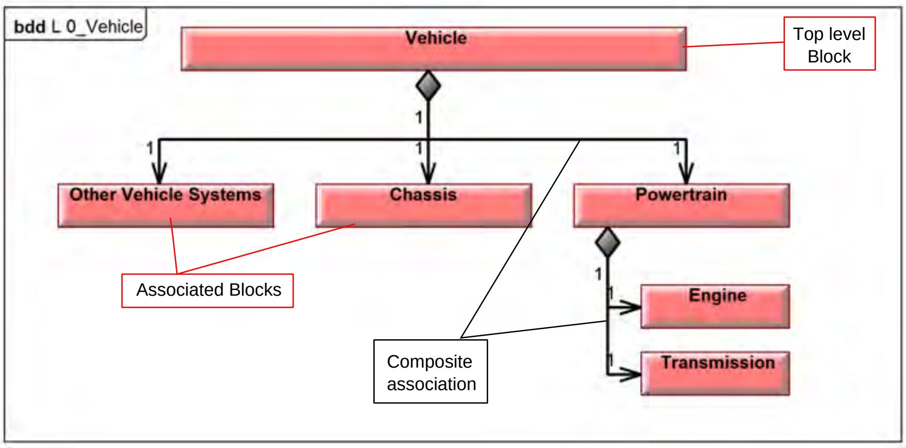

Figure 2.7: Block Definition Diagram for the structure of the system

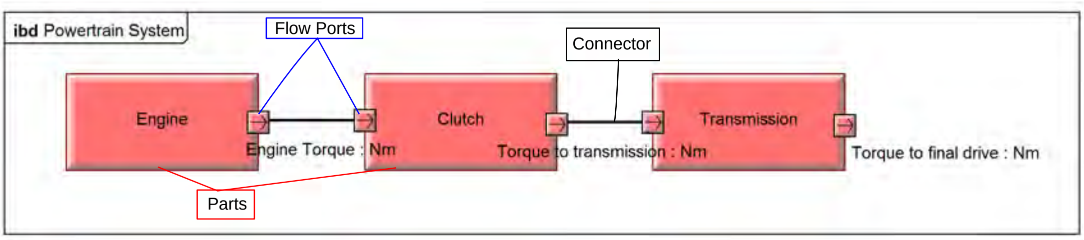

Figure 2.8: Internal Block Diagram

which the related variables are block property values. In Figure 2.9 is shown an instance of parametric diagram including constraint formula, constraint parameters and the related block property values.

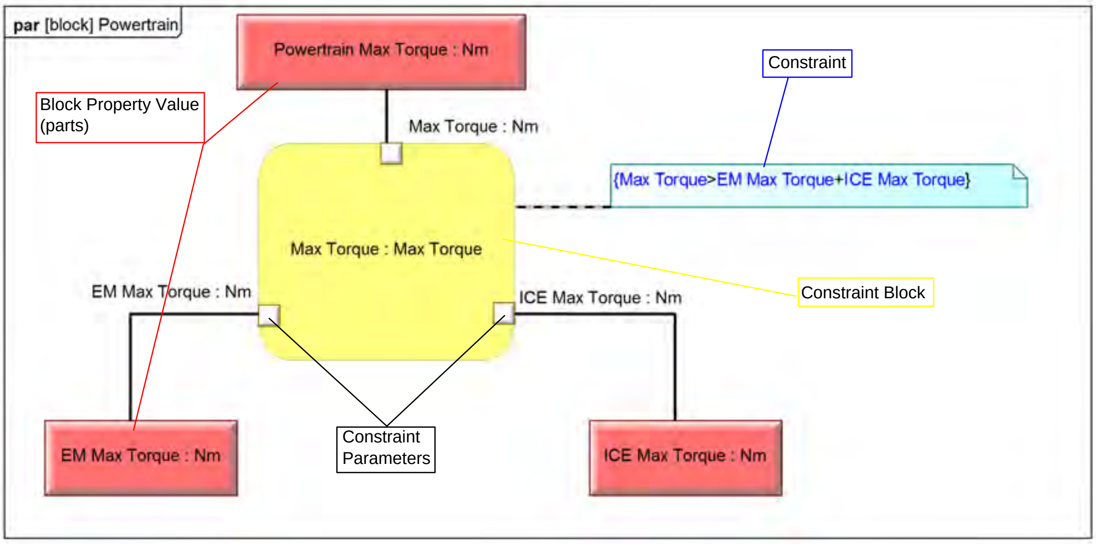

Figure 2.9: Parametric Diagram

In every diagram we can use requirements for a better definition of blocks, activities, etc. and requirements are the central modeling item in fact the requirements from one side decide the functions of the system and then of the subsystems and components and from the other side describe the parameters, the performances, etc. of them. In Figure 2.10 shown a typical requirement diagram.

Finally, in in Figure 2.11, there is an overview of the SysML^{TM} window. Here there are three pane: the explorer panes that provide access to the various items (parts, activities, etc.), diagrams and relationships within the model; the output panes group provides a set of individual pane which show for example the contents of a selected item in the explorer pane or the results of a model item search; the property pages that show properties, description, links and other about a selected item. At the end there is the diagram pane in which are open, build and modify the diagrams.

More information about System Engineering and SysML^{TM} can be found in Atego, *SysML Tutorial v8-0-0.pdf*, *INCOSE*, F. Sanford, *A Practical Guide to SysML: The Systems Modeling Language*, Fowler, *UML Distilled: A Brief Guide to the Standard Object Modeling Language*, OMG, *OMG Systems Modeling Language (OMG SysML^{TM}) Version 1.3*, OMG, *OMG Systems Modeling Language (OMG SysML^{TM}) Tutorial 19 June 2008*, *OMG Systems Modeling Language tutorial and specification*, *Software articles*.

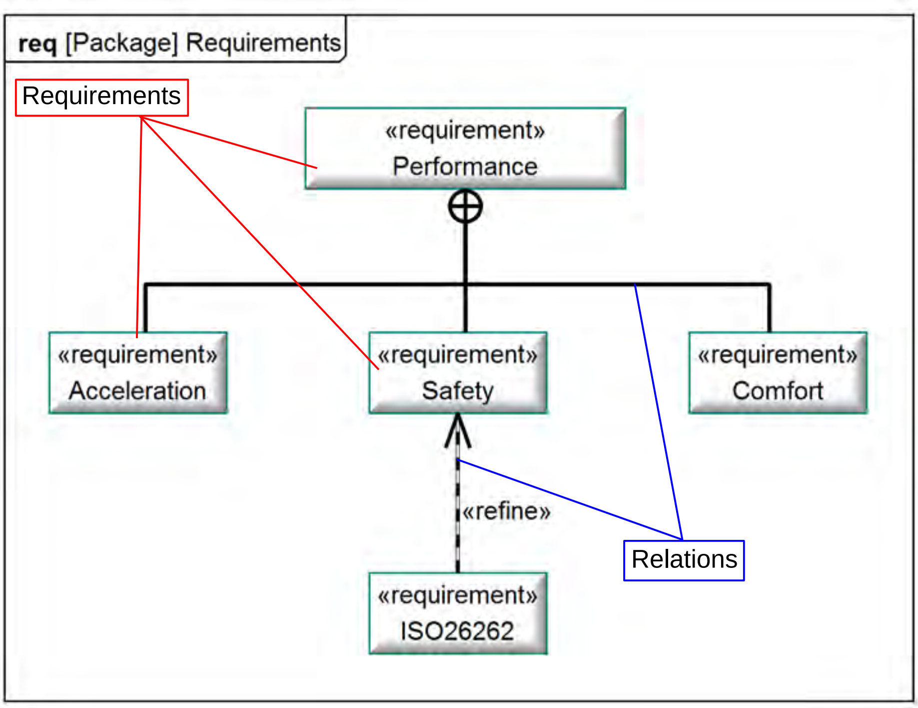

Figure 2.10: Requirement Diagram

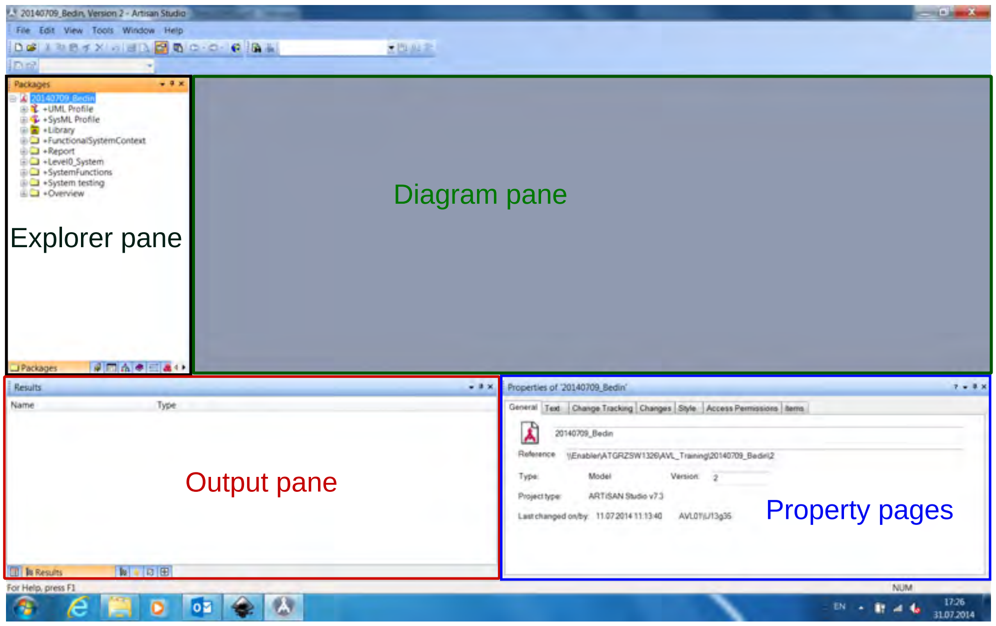

Figure 2.11: SysML^{TM} overview

3 Methodology - RQs cascading workflow

3.1 Methodology overview

The methodology that has been applied is based on the current AVL RQs engineering approach and is adapted to E/E components: RQs are cascaded through the levels and after components evaluation phase, they attributes are available. RQs cascading workflow, represented on Figure 3.1 and integrated with other phases that complete the total E/E development process (these phases are physical layer specification, signal definition, etc.), is a schematic view of the methodology applied and explained in the following sections.

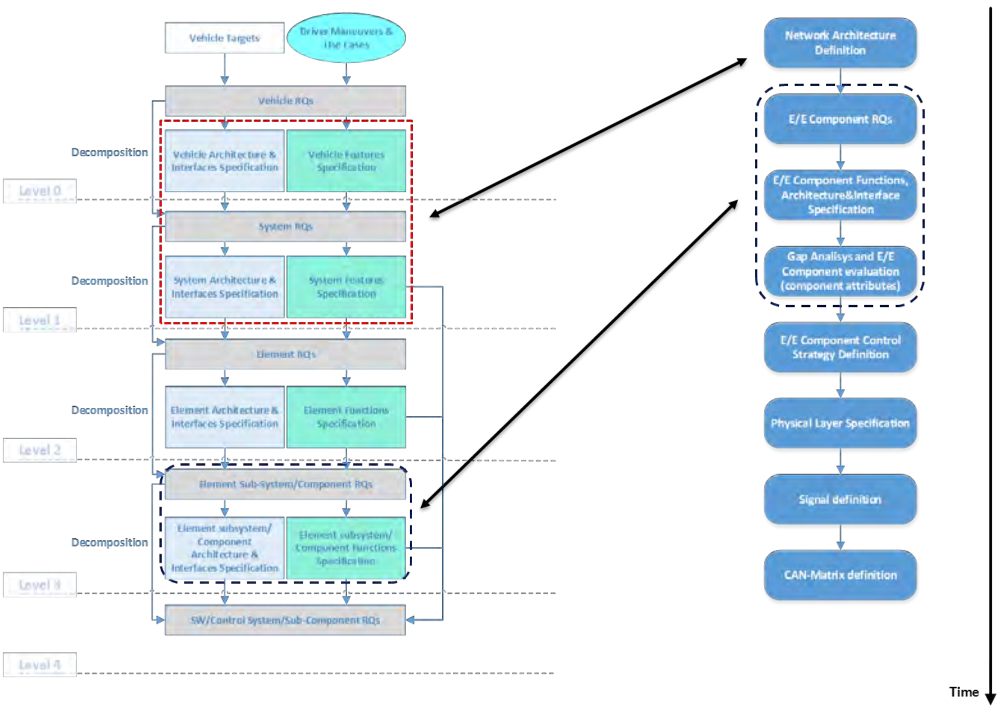

Figure 3.1: E/E development process

Then, a parallel path is added, to show the integration process with different kind of simulation tasks and tools, depending on the level in which simulations are done (see Figure 3.2).

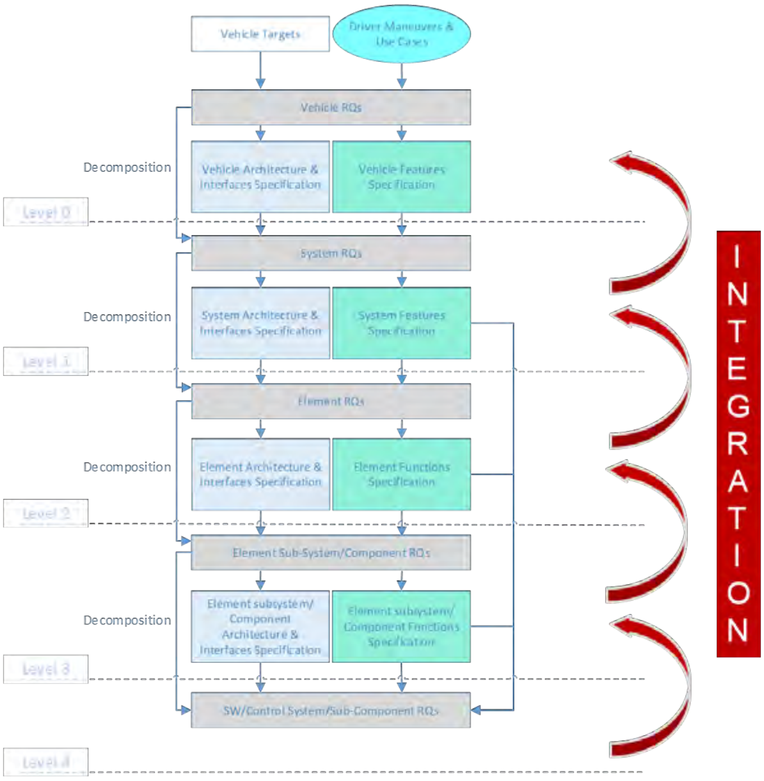

Figure 3.2: RQs cascading workflow and integration process

3.2 RQs cascading phases

3.2.1 Vehicle - Level 0

Input for RQs cascading are from one side vehicle targets, costumer wishes, legislative constraints, environmental conditions, safety regulations and from the other side driver maneuvers like brake, creep, launch, standstill, and drive & accelerate. From these, level 0 RQs are derived. Initially, level 0 RQs are derived from one side from vehicle targets, e.g. from fuel economy target a RQ like *"Hybrid vehicle shall have a fuel consumption less or equal to 20% respect the conventional vehicle"* and from the other side other RQs are derived from use cases. Then, from RQs, vehicle features (related to use cases) and vehicle architecture are derived and specified in order to fulfil them.

Vehicle features specification mainly consists on:

- the definition of a sequence of sub-features that shall be done to provide the features;
- the definition of the allowed transitions between features and sub-features;
- the definition of feature inputs and outputs.

Vehicle architecture specification consists on:

- definition of systems;
- features allocation to vehicle systems like powertrain, body and chassis (e.g. the vehicle feature "stationary force consumption" is related to the driver maneuver "brake" and is allocated to both PT and chassis);
- definition of mechanical, thermal, electrical, etc. interfaces between systems (the amount of details in the architecture depends on how much detailed are the features).

Usually vehicle architecture, and then vehicle systems, is already defined. In fact vehicle systems do not depend on vehicle type (i.e. every vehicle has a chassis, a body and a PT). Looking at E/E and at the E/E development process in Figure 1, at this level start the network topology definition and the definition of general E/E RQs like operating voltages, working temperature range, resistance to vibration, EMC Standard, etc. These requirements can be directly cascaded to components and become boundary conditions for their evaluation. In Figure 3.3, some example of these E/E RQs. The network topology definition is done considering the existents vehicle networks and vehicle CUs (in case the project starts from an already existent vehicle), the systems that shall be linked by network, flash and diagnostic concept, type of network like CAN, LIN, FlexRay, etc.

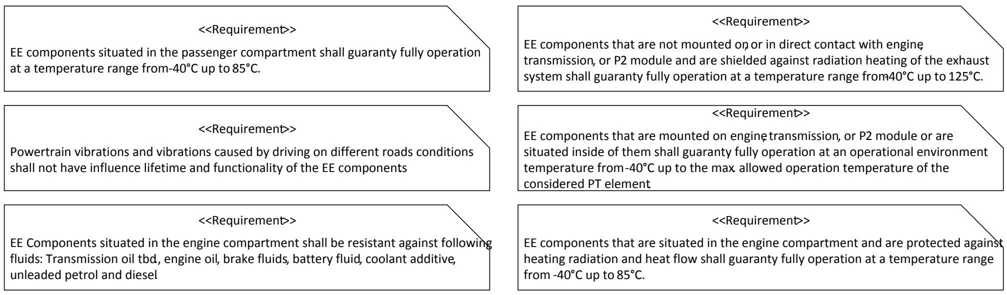

Figure 3.3: Level 0 E/E requirements

Vehicle features and architecture specification become inputs for lower level RQs.

3.2.2 Systems - Level 1

From level 0 specifications (features and architecture), system RQs can be derived. Then, for every system (PT, chassis, etc.), features and an architecture are specified to fulfill system RQs like is done for vehicle level. Vehicle RQs that do not lead to features, can be directly cascaded to vehicle system looking at what system attributes affect the considered RQ. Finally, system features and some interfaces become input for software (SW) RQs (level 4) and for level 2 RQs. It is important to note that during RQs cascading, in order to provide the functions that allow fulfilling RQs, new elements/components can be found. Every new element/component affect both its own level RQs and other levels RQs.

3.2.3 Elements - Level 2

Going down to level 2, element RQs are defined from level 1 specification. Then element functions and architecture are specified again repeating the same steps. At this point, before to go to level below some considerations on element (element evaluation) have to be done, otherwise is not possible to completely define the next architecture. These considerations lead to component sizing and may lead to the need of other components to help to deliver a defined function. Furthermore, lead also to an update of the architecture, included the interfaces. From element specification other SW RQs can derived.

3.2.4 Components - Level 3

In level 3, components are found depending on system structure. Again, the same steps are repeated starting to derive level 3 RQs and the consideration done for level 2 elements, allow defining level 3 components. Again also for components, several considerations can be done and other components, functions and RQs derived. In level 3 are listed also requirements for terminal, wiring and connectors (e.g. number of mating cycles). At the same time at this point start also E/E component evaluation and specification in according with the added phases of the E/E development process.

3.2.5 SW & Sub-Components (Control Systems) - Level 4

As for the levels above, the procedure is applied to level 4. Usually it is not much used in system modeling because it contains RQs for CUs functions and software development. And as look at the previous sections, level 4 RQs derive also for levels above specification, which are used for SW development.

4 Example of RQs cascading workflow application using SysML^{TM}

To better understand the process, several examples are taken under consideration. Note that the used features are very simplified to keep a good level of clearness and only electrical energy and information interfaces are considered and then represented.

4.1 Vehicle - Level 0

4.1.1 Stationary force consumption

The first considered vehicle feature is Stationary Force Consumption, which is related to the driver maneuvers Brake. This feature, for a conventional vehicle include only conventional braking, while for a hybrid vehicle include both conventional braking and recuperation. Stationary force consumption, represented on Figure 4.1, is composed by several sub-features and an activity diagram, is used to represent it. Feature decomposition is important to well understand the role of a system on the feature. Looking at the considered feature, it is provided from both PT and chassis. Then, its decomposition to sub-features is necessary to assign them to an only one system (e.g. conventional braking to chassis and recuperation to PT).

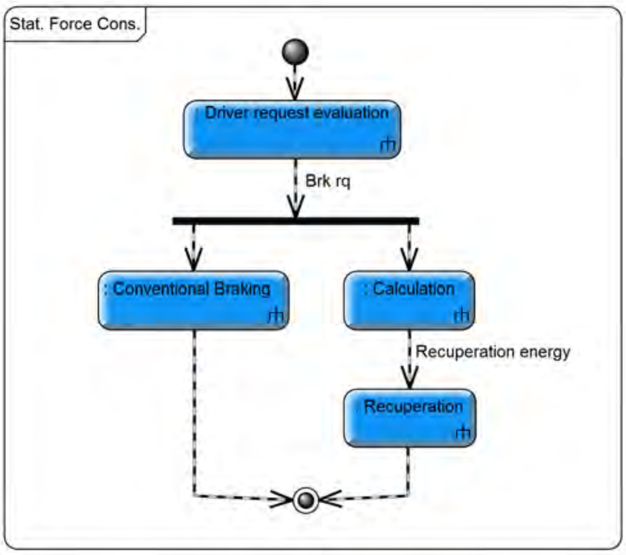

Figure 4.1: Stationary Force Consumption (SysML^{TM} activity diagram)

At this level the architecture is the same for every kind of vehicle. It contains the following systems: chassis, body, PT, and some E/E that in this case are low voltage board net (LVBN), high voltage board net (HVBN) and Hybrid Control Unit (HCU) because they are involved on the feature. An internal block diagram (ibd) is used to represent the architecture. Now, every sub-feature shall be related to the system that provide it:

- driver request evaluation is provided from body system because it contains all the HMI components (brake pedal, acceleration pedal, driver screen, etc.);
- conventional Braking is provided from Chassis (it contains the brake system);
- calculation in case of recuperation is assigned to HCU;
- recuperation is provided from PT.

The arrows that link the sub-features represent the information flow and these became interfaces between systems: i.e. body and chassis shall have a common information interface, called Brk rq (brake request). The same is true for body and HCU. In Figure 4.2 the derived architecture is shown.

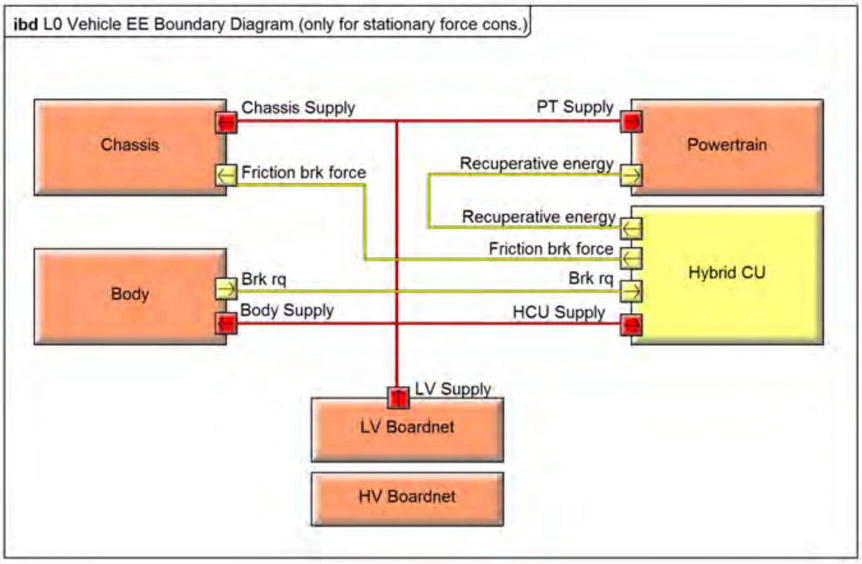

Figure 4.2: Level 0 Architecture (ibd SysML™)

Obviously not only information (the yellow flow ports) and electric energy (the red flow ports) interfaces are in the architecture, but the method is applied to E/E and then only these stereotypes[1] of interfaces are considered and used.

4.1.2 Stationary force generation

The same, has been applied for Stationary Force Generation vehicle feature and the results is shown in Figure 4.3.

Note that there is not an internal block diagram for every feature, but all the interfaces between systems, resulting from the assignment phase, are collected in a single diagram.

4.2 Systems - Level 1

First, vehicle RQ about fuel consumption is cascaded to vehicle systems looking at what system attributes (e.g. weight, efficiency) affect vehicle fuel consumption. Then the other examples regard to system specification (architecture and features specification). From architecture (specification) some system RQ can be derived:

- LVBN shall have a power supply interface with all vehicle systems;

[1]Stereotypes are defined in AVL system modeling rules (e.g. E/E stereotypes consist on yellow blocks for E/E components, yellow flow ports and connectors for information interfaces and red flow ports and connectors for electric energy interfaces).

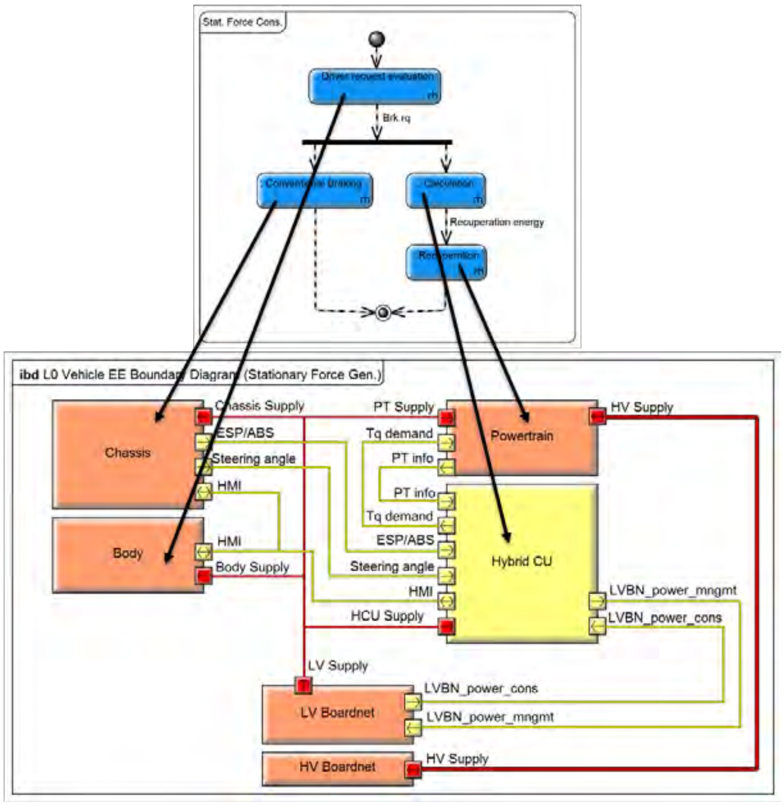

Figure 4.3: Assignment of functions to vehicle systems

- Chassis shall execute the requested brake force;
- PT shall support recuperation feature;
- HCU shall calculate recuperative energy;
- HCU shall communicate the recuperative energy to PT.

For body, the involved HMI component in the feature is the brake pedal, so the requirements can be directly referred to it becoming level 3 RQs:

- Brake pedal position shall be on percentage (derive from the need to deliver recuperation);
- Brake pedal and HCU shall have a common interface called Brk rq;
- ...

From these, the following RQs are considered:

1. Chassis shall support conventional braking;
2. PT shall support electric driving, boost and start & stop feature.

4.2.1 Direct RQ cascading - fuel consumption

Considering the RQ *"fuel consumption shall be less than 20% respect conventional vehicle fuel consumption"*, it can be directly cascade to systems. In order to derive level 1 RQs from it, the attributes of every systems shall be analyzed to define how they contributes to fulfill the RQ. In Figure 4.4, for every vehicle system are listed the attributes that affect vehicle fuel consumption. For instance looking at the drag coefficient, assigned to body, it has to be decrease in order to achieve a better fuel economy.

Then, level 1 RQs could be:

- PT weight shall be less or equal to 400 kg;
- LVBN power consumption shall be less than 400 W;
- Rolling coefficient shall be less than 0.009.

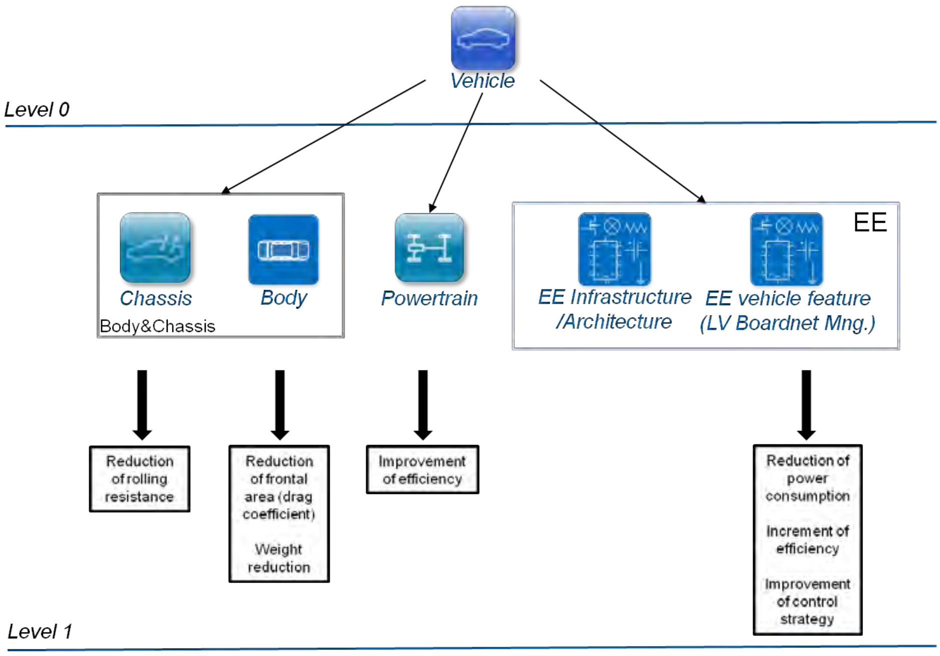

Figure 4.4: Fuel consumption RQ from level 0 to level 1

4.2.2 Conventional braking

From chassis considered RQ, the need to provide conventional braking is derived. First is important to define this system feature, which is composed by a list of functions, and to define the architecture (level 2 elements of chassis). Second, functions are assigned to elements. The activity diagram of feature, the system architecture and the assignment phase are shown on Figure 4.5.

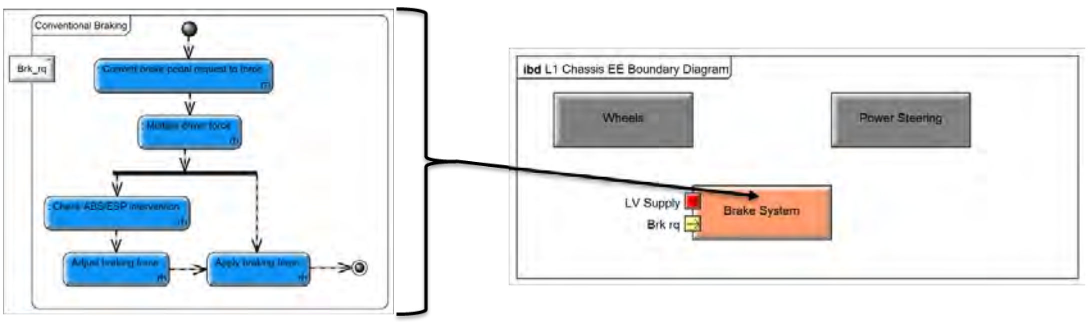

Figure 4.5: Conventional Braking and chassis architecture specification

Then from features inputs and outputs, from information flow and from other chassis RQs, architecture can be better specified (what elements and what interfaces between them). In this case, all the functions of conventional braking are related to Brake System and its components, so that these are directly linked to L3 components.

4.2.3 Electric drive, boost and start & stop

These functions are taken under consideration looking only at PT system and its elements. As for conventional braking, these shall be specified very well in order to better specify the interfaces (architecture). The result is new three lists of functions (very simplified to understand the procedure) (see Figure 4.6).

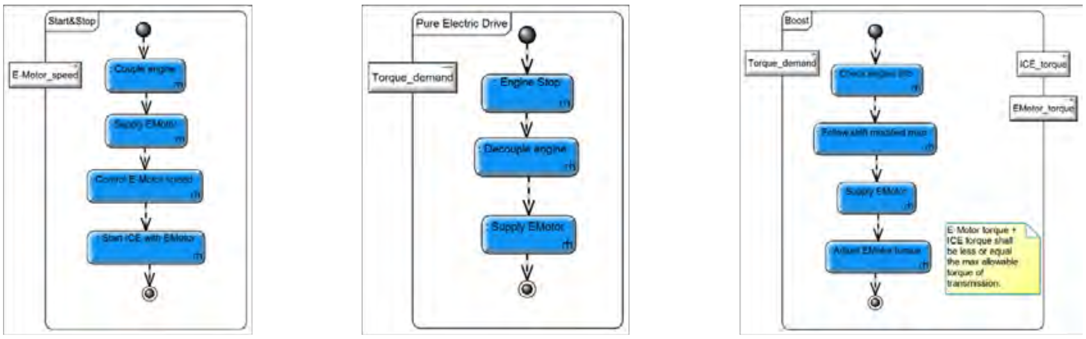

Figure 4.6: Electric driving, boost and start & stop activity diagram

As this example really simplified, the activity diagrams are not complete and referred only to PT. In fact several other input are also needed and are included on Calculation function provided for instance from HCU, HMI, sensors etc. (see Figure 4.7). Some of these can be: high voltage battery state of charge (SOC), LVBN power consumption, vehicle speed, acceleration pedal position, ABS/ESP intervention, E-Drive button activation and so on. At the same time, also output like information to send to transmission, engine or driver screen are leaded to interfaces. This means that a features shall be more detailed as possible in order to define immediately all the interfaces between elements. Now, after that functions are assigned to PT elements (e.g. couple engine to E-Drive, engine stop to EMS that is considered part of engine system, supply E-Motor to HV Battery and E-Drive), the architecture (with interfaces) are specified and the results, modeled as ibd, is reported on Figure 4.7.

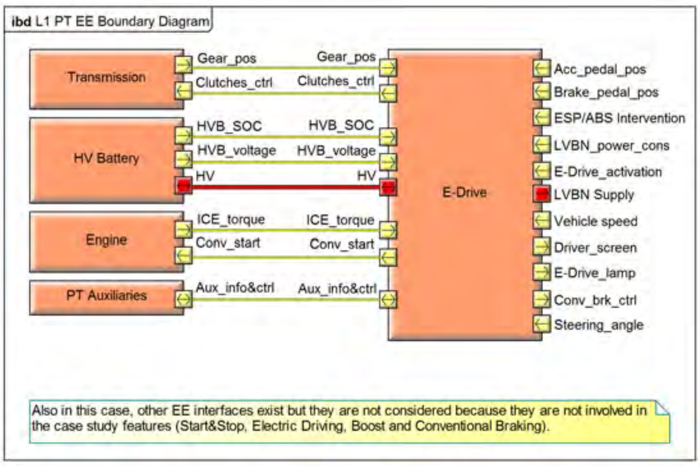

Figure 4.7: Powertrain architecture

4.3 Elements - Level 2

System specifications described in Section 4.2 are used to derive level 2 RQs, in particular conventional braking from one side and couple/decouple engine from the other are used to derive Brake System and E-Drive RQs. Finally, is represented how RQ reduce fuel consumption is split to PT elements.

4.3.1 Conventional braking

From architecture, the following RQs can be derived:

- Brake system shall convert brake pedal pressure to braking force;
- Brake system shall have a power supply interface connected to LVBN (this derive from the need to supply ABS/ESP component);
- Brake system shall support conventional braking.

Obviously these are only a part of the big amount of RQs that usually are derived. As already said, conventional braking feature is directly derived to brake system elements. Applying the same procedure, feature and architecture are specified and this is summarized in Figure 4.8.

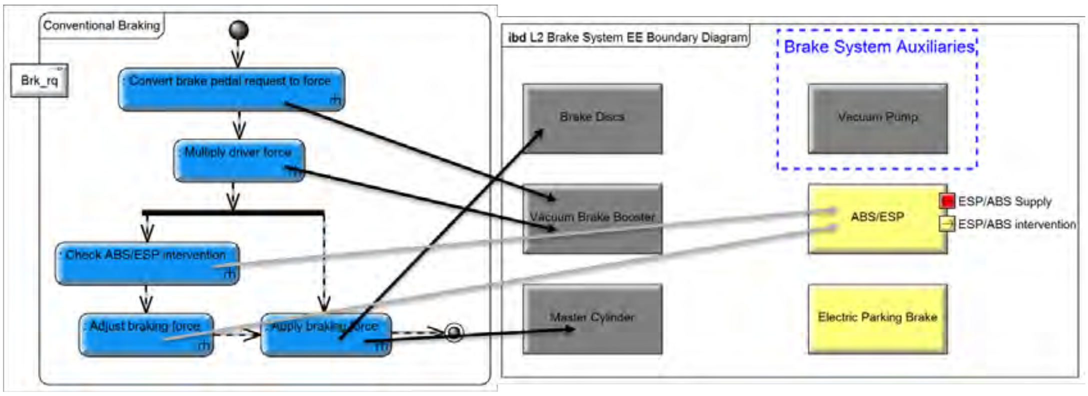

Figure 4.8: Brake system functions and architecture specification

Some brake system components shown in Figure 4.8, are represented mounted together on Figure 4.9.

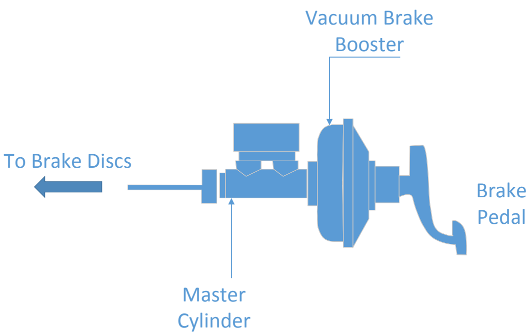

Figure 4.9: Brake system components

Consideration

At this point, some considerations can be done:

1. the interface between brake pedal and vacuum brake booster (Brk rq) is a mechanical interface and not an information interface due to brake pedal is directly mounted on vacuum brake booster and between them are exchanged pressure/force quantity.
2. vacuum brake boost functions (convert brake pedal request to force and multiply driver force) are possible only under certain conditions: vacuum brake boost work properly only if the absolute pressure (vacuum) inside it remain included a certain range. This has particularly influence from a safety point of view and become a RQ for vacuum pump (level 3 RQ).

3. in a hybrid vehicle, in which Pure Electric Driving is actuated (i.e. ICE off), vacuum pump cannot be mechanical. In fact, when engine is off also it is off and the vacuum inside vacuum brake booster could not match the necessary vacuum level needed to actuate the previous mentioned functions. So that vacuum pump shall be electrical. That means that it shall be LV supplied and shall be controlled in some way.

Architecture update

After these considerations, brake system boundary diagram can be updated with the new information: the block Vacuum Pump, which before was mechanical component (hardware component is stereotyped with a gray block), become an E/E block (E/E component is stereotyped with a yellow block)

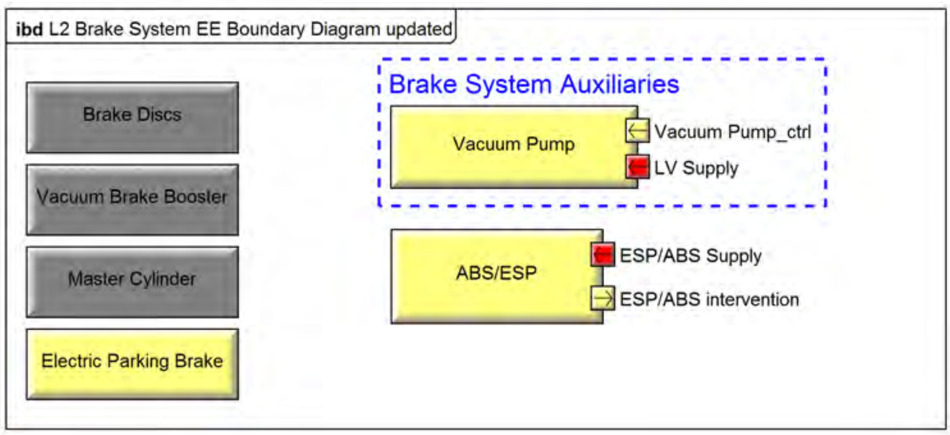

Figure 4.10: Level 2 architecture for Brake System

4.3.2 Couple/decouple engine

From level 1 specification, and assuming that CUs are connected together using a CAN network (this derives from network topologies investigation done at highest level), the following RQs can be derived. Obviously, there are not only level 2 RQs but also lower lever RQs (e.g. RQs leading to CUs are level 3/4 RQs).

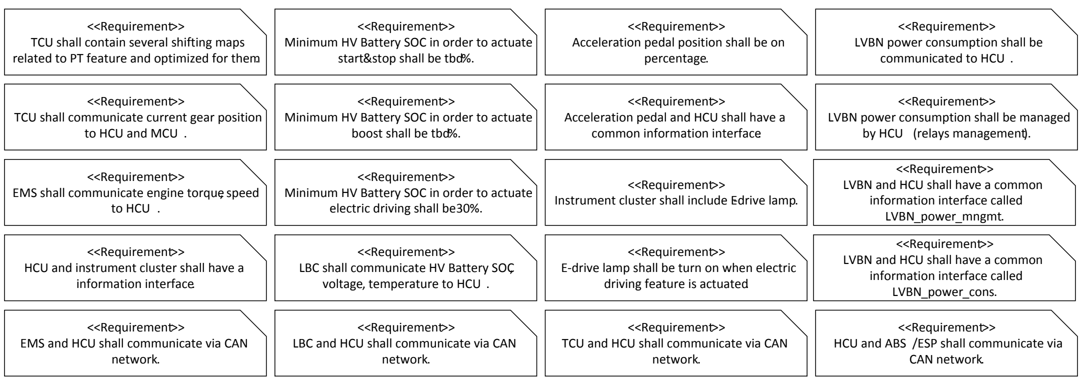

Figure 4.11: RQs from level 1 specifications

Now, focusing on "couple/decouple engine" functions, these are linked to E-Drive. But E-drive is only a system view and not a block/component (is a group of blocks) and these functions can be allocated to E-Drive components. So that we have to define the architecture of E-Drive and looking at the considered functions, a separation element is needed (new component

RQ). E-Drive architecture with the interfaces derived from the considered features is shown in Figure 4.12.

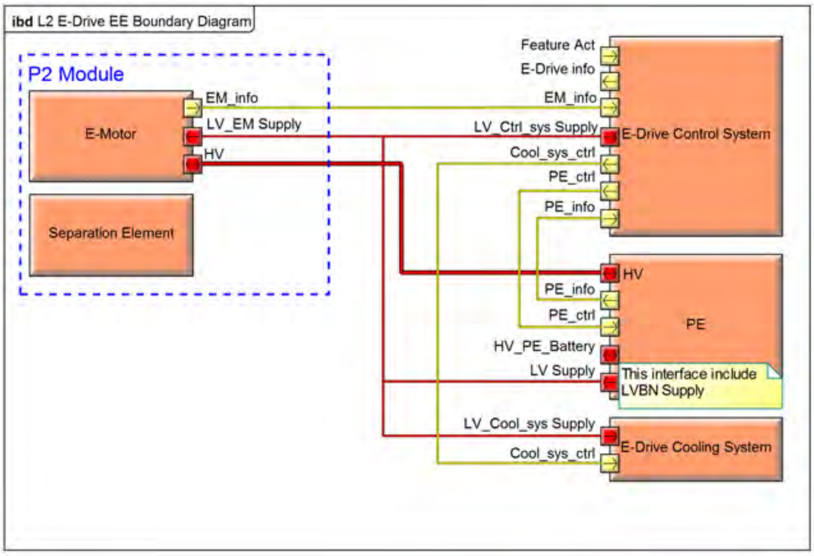

Figure 4.12: E-Drive architecture

4.3.3 Fuel consumption RQ

As for level above, also in this case RQ is directly linked to level 2 element attributes. Considering only PT block, Figure 4.13 shows how element attributes affect fuel consumption and then level 2 RQs can be derived.

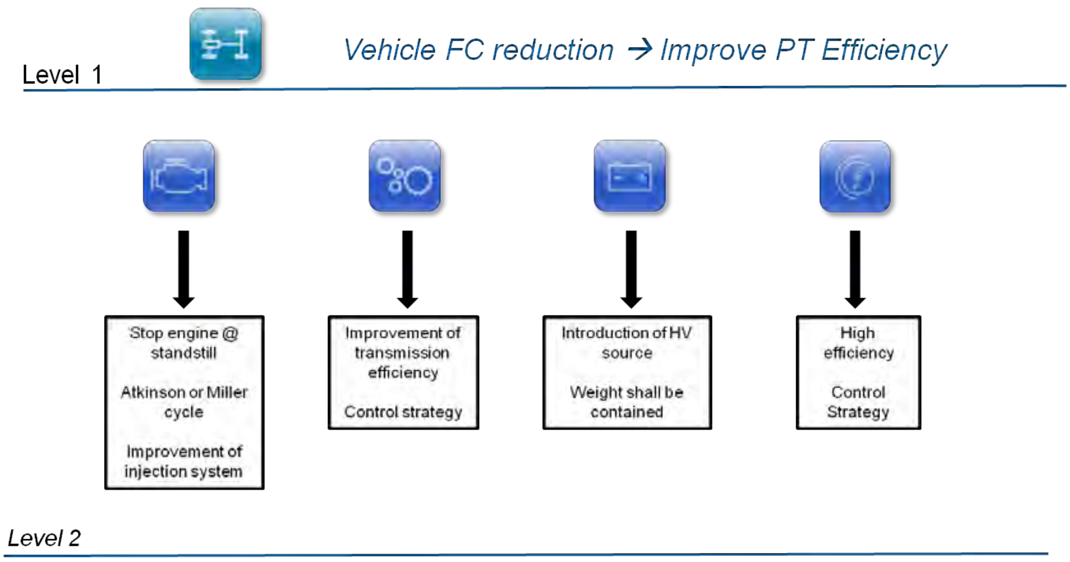

Figure 4.13: Fuel consumption RQ from level 1 to level 2

To fulfil PT RQ *"PT efficiency shall be tbd%"* or *"PT weight shall not exceed 400 kg"*, element RQs are derived like: *"Engine shall be turned off during vehicle standstill"* or *"HV Battery weight shall be less than tbd kg"* and so on.

4.4 Components - Level 3

In this case, vacuum pump and separation element RQs are derived. For fuel consumption RQ, the procedure is always the same. For instance, the level 2 RQ *"HV Battery weight shall be less than tbd kg"* is cascading to HV Battery components (level 3/4) and become *"HV Battery cooling system weight shall be tbd kg"*.

4.4.1 Components RQs and components evaluation - vacuum pump

From level 2 specification, the following RQs can be derived.

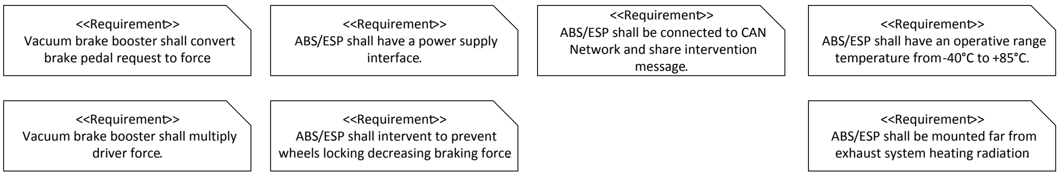

Figure 4.14: Brake system components RQs

The RQ *"ABS/ESP shall be connected to CAN network and share intervention message"* derive from necessity to communicate to Transmission Control Unit (TCU) to avoid gear shifting and to Engine Management System (EMS) to avoid engine torque increase during ABS intervention. Considering only the vacuum pump, after that its RQs have been derived, its design parameters shall be defined and its control method shall be selected. To size correctly vacuum pump, vacuum brake booster data have to be available. In particular way the volume to evacuate, the estimated leakage related to pressure, minimum and maximum abs. pressure (upper and lower range limit). With this data is possible to select the pump looking at its average pump suction capacity. To define this parameter, the formula (4.1) can be used. S is the average pump suction capacity expressed in m^3/h, V is the volume to evacuate in m^3, t is the estimated evacuation time in h and P_0, P_1 are respectively the upper and lower limits of the pressure range (absolute pressure).

$$S = \frac{V}{t} ln\left(\frac{P_0}{P_1}\right) \tag{4.1}$$

Vacuum pump can be open loop or closed loop controlled. The difference is that in the first method the pump is controlled following a certain maps estimated initially, in the second there is a pressure sensor mounted on vacuum brake booster and with its pressure information the pump is switched on and off (hysteresis control). Usually, in case of pressure sensor failure, pump control strategy change to open loop control. In this case closed loop control is selected and a new component is defined: a pressure sensor to measure pressure inside vacuum brake booster. Furthermore also a RQ for noise (derived from vehicle NVH target) are derived for vacuum pump, in fact usually a vacuum pump is very noise component due to compression and expansion waves (pressure bangs) and this can be more notice from driver when engine is turn off (electric drive). Not only vacuum pump characteristic is important, also its parameters that can affect vehicle targets shall be analyzed to derive it RQs. These are:

- weight, that impact on the total added weight due to hybridization;
- power consumption that affect LVBN power consumption and then vehicle fuel consumption (vacuum pump is a heavy consumer but it runs few times, its power consumption depends on how many times the driver brakes, on the quality of brake boost to maintain the vacuum and on the pressure range);
- noise that affects NVH RQs;
- lifetime that shall be aligned to durability RQ.

After these considerations, brake system boundary diagram can be updated with the new information and other RQs can be derived:

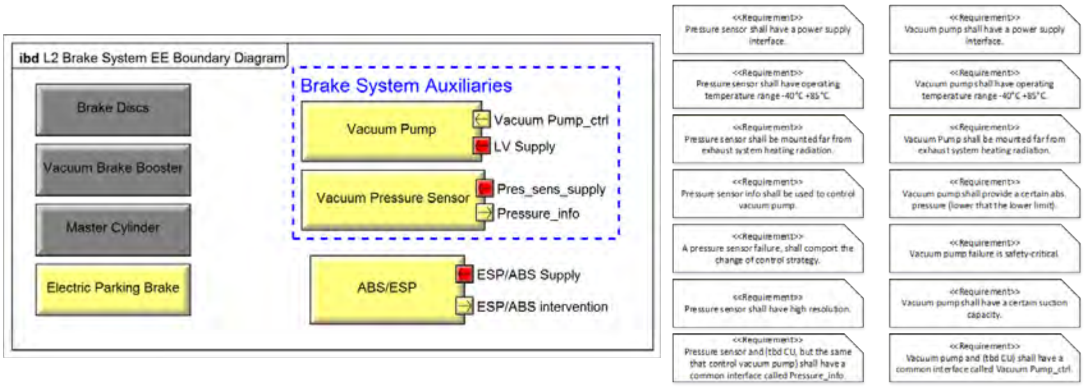

Figure 4.15: Vacuum pump and pressure sensor RQs

4.4.2 Components RQs and components evaluation - separation element

Before to define its typology, its requirement shall be derived (see Figure 4.16).

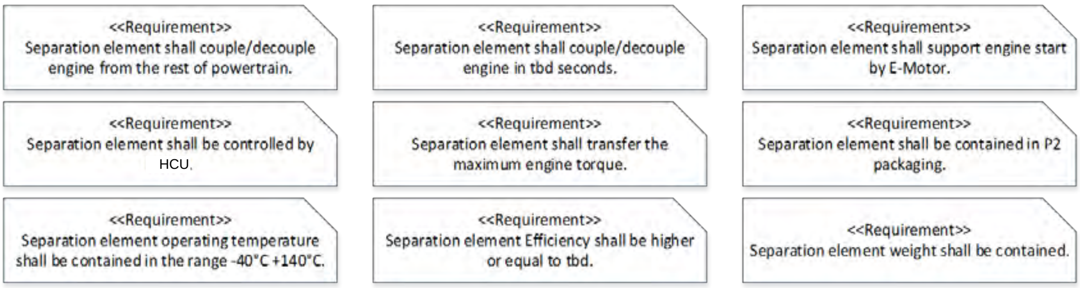

Figure 4.16: Separation element RQs

The RQ *"Separation element operating temperature shall be contained in the range -40°C +140°C"* derive from P2 components temperature range (it is derived from the position of P2 module in the powertrain). Separation element RQs are the starting point for its determination. There are different type of separation element like a clutch. Furthermore, for every kind there are several sub-typologies: for example dry or wet clutch, single- or multi-plate clutch, mechanical- or hydraulic-actuated clutch and so on. Some considerations about them could be: wet clutch allows to transfer bigger torque than dry clutch (with same size) or multi-disc clutch is smaller than single-disc clutch but at the same time it is more expensive. All the typology shall be evaluated and then compared together looking at the RQs and the choice need to be confirmed by customer. In this case, there is an additional RQ: separation clutch shall be the same of Dual Clutch Transmission (DCT) clutches (vehicle transmission is DCT). So that is not necessary to define a new element. DCT clutches are single-plate wet clutches and obviously they respect the RQ about the max transferred torque (they are derived from conventional vehicle). Now that we know the type, main data are available and it is possible to define its control method. Main functions of separation clutch are couple and decouple engine but for start & stop feature during drive, it has to be controlled in slip mode (clutch discs rotate at different speeds) and this comport a new RQ about the max allowable delta-speed in slipping mode (e.g. 2000 rpm). Slip is necessary to transfer energy from E-Motor to ICE and to wheels at the same time (also DCT clutches are slipping during this feature and it leads to new RQs for DCT and TCU) without that driver feels what is happening. Regarding clutch control method, it depends on clutch characteristic: we want that the clutch transfers engine torque and the torque transferred by a wet clutch is approximately proportional to oil pressure. Furthermore, wet clutch shall be lubricated. From these considerations derive the need of element that supply oil for clutch actuation and lubrication and elements that control the oil flow.

Finally, also clutch closing time is important and it is chosen looking at:

1. a fast actuation, that means:
   - lower losses (less slip) i.e. lower heat production and then higher lifetime;
   - and high response;
2. not too fast because:
   - get worse the drivability;
   - can badly affects the functionalities of the rest of powertrain (e.g. the DMF);
   - oil pump works at high speed, i.e. high losses and then heat production and low efficiency.

So that, several variables shall be considered when RQs for closing time are formulated. Then new RQs for clutch are formulated:

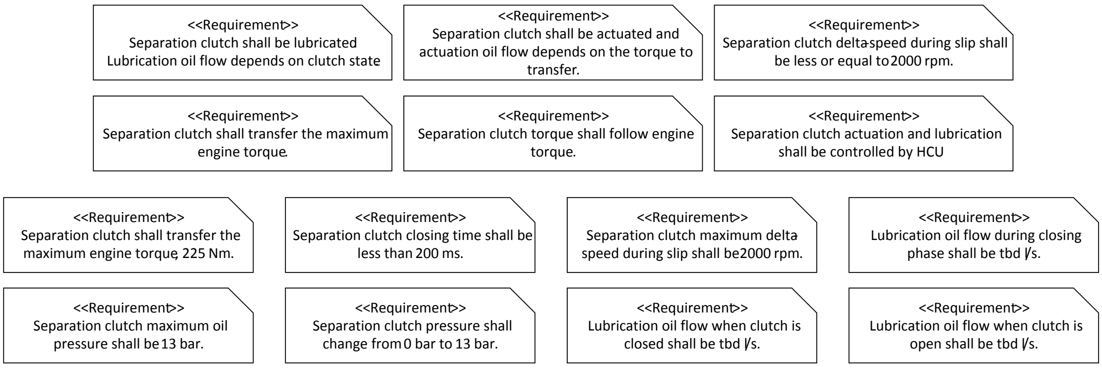

Figure 4.17: Separation clutch RQs

Separation clutch control

Starting from requirements of separation element, new functions are derived to control this component. In order to control oil flow for lubrication and actuation, 3-ways solenoid are needed. If the control strategy is based on close-loop, also pressure sensors are needed (one on the clutch because clutch pressure is proportional to torque to be transferred; one on the line in order to control max pressure in the circuit and the lubrication is obtained for difference with clutch pressure). At the same time, an oil flow supplier is necessary (oil pump). One solution could be to use the already existent oil circuit of transmission and replace the current mechanical oil pump with a new one with a higher power, but this means also modify the oil circuit. Finally, there is a customer request: P2 module shall be independent on vehicle model and reusable in future, and then new electrical oil pump and hydraulic circuit are added. The new functionalities are then allocate to components as is shown in Figure 4.18.

Then, level 2 architecture can be updated and the level 3 architecture derived (see Figure 4.19).

Furthermore level 3 RQs for electric oil pump (because is a level 3 component) can be derived (Figure 4.20).

Now, with RQs is possible to select the component.

4.4.3 Components RQs and components evaluation - electric oil pump

Regarding electric oil pump, its two main characteristics are: oil flow provided and time to build up the pressure up to the maximum value (this affects clutch closing time and at the same time has negative impact on pump power). It is speed controlled because its speed is linked to the provided oil flow and then the line pressure value. Knowing clutch attributes, is possible

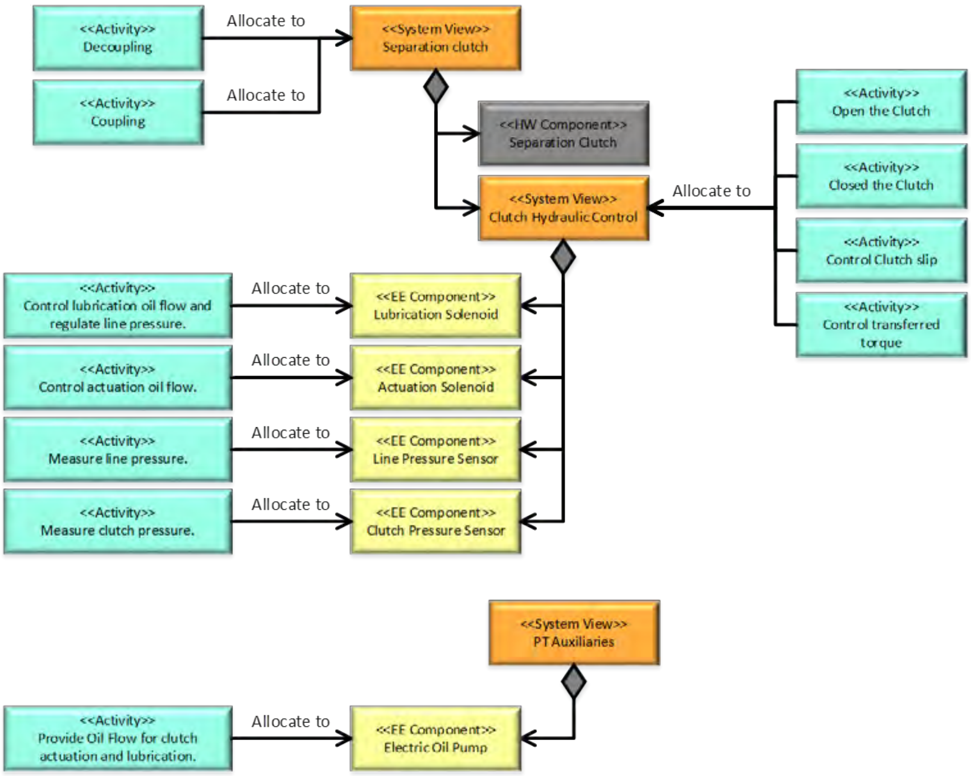

Figure 4.18: Separation clutch control functionalities

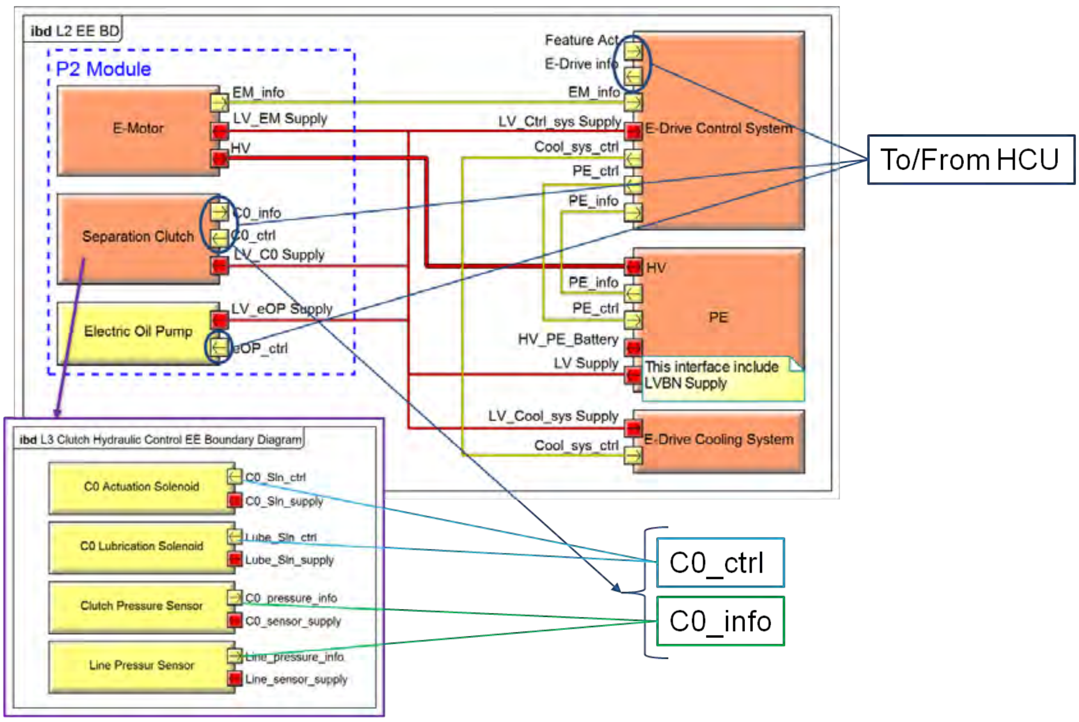

Figure 4.19: Level 2 and level 3 architecture and interfaces

estimate with a good approximation the theoretical hydraulic power (without efficiencies) of electric oil pump. Separation clutch is modeled like a piston and a spring and the data used refer to oil at 90°C. With the following assumptions:

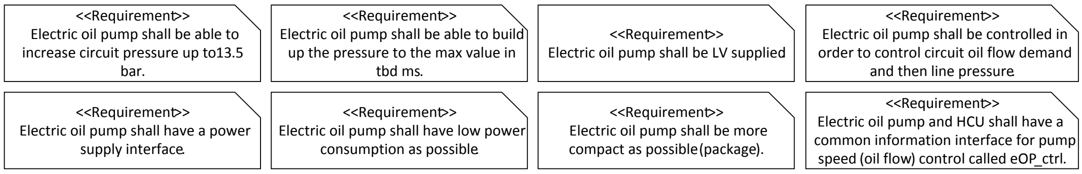

Figure 4.20: Electric oil pump RQs

- pipe diameter: 1 cm;
- offset: 0,5 bar;
- closing time: 200 ms *(this is very important: halve closing time cause double pump theoretical hydraulic power)*

component data and under normal conditions, oil pump theoretical hydraulic power shall be greater or equal to $77.6W$. Usually oil pump power is greater than the one calculated because the oil pump shall be able to increase line pressure very fast to fulfill clutch closing time RQ and mechanical, volumetric and electrical efficiency shall be considered. However, is possible optimize power consumption, using different control methods. Is important to observe that electric oil pump is a high power consumer and that affect LVBN power consumption and then vehicle RQs and targets. Its evaluation shall be done with criteria.

4.4.4 Components RQs and components evaluation - solenoids and pressure sensors

For the considered examples, clutch control components are allocated to level 4, but depending on the definition of the system structure, they can be allocated to level 3. Looking at their functions, represented in Figure 4.18, their RQs can be derived:

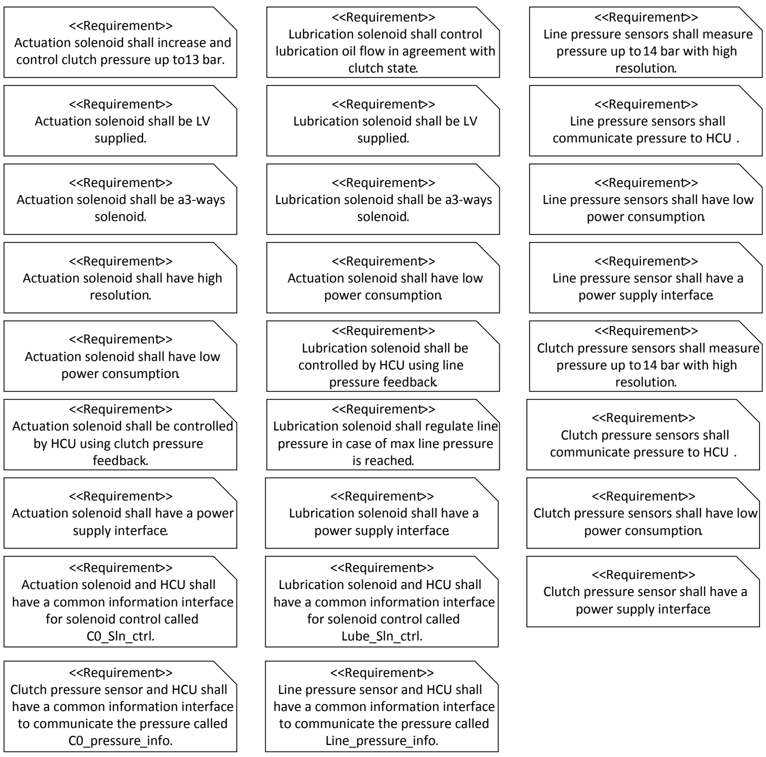

Figure 4.21: Solenoids and pressure sensors RQs

From RQs, components can be evaluated and selected. Following some considerations about them.

Pressure sensors

In general, for pressure sensors, the following aspects are true:

- their weight is negligible (very low, does not affect vehicle weight RQs);
- their power consumption is negligible (usually current supply is some tens of mA thus very low, does not affect significantly LVBN power consumption);
- their output is a voltage proportional to the pressure, and it is always lower than input voltage;
- usually there are 3 terminals: one for input voltage, one for output voltage and one that is the common (see Figure 4.22).

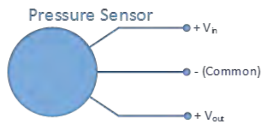

Figure 4.22: Schematic representation of a pressure sensor

Solenoids

Looking at the actuation solenoid, it has to control the oil flow to separation clutch because torque transferred by clutch depends on its pressure (and then on oil flow). In this case, clutch open (i.e. engine decoupled) means no oil flow to the clutch and then actuation solenoid closed. In order to minimize the LVBN power consumption when ICE is off (worst case because HV battery shall supply all the LV components and the E-Motor without ICE support) actuation solenoid shall NC (normally closed) solenoid. This means that it is closed without be powered (no current means no power consumption). For the same reason, lubrication solenoid shall be a NO (normally open) solenoid (when clutch is open, needs more lubrication, and solenoid is open with zero or low current and provides the needed lubrication oil flow, whereas when clutch is closed lubrication flow is very low and the solenoid needs higher current to be closed). For that above, when clutch is open, solenoids power consumption is very low. Finally, in order to protect the pump from high line pressure, a pressure regulator is needed, but without add other components (i.e. increase the cost and the weight), lubrication solenoid is also used to control line pressure. For solenoids, the following considerations can be done:

- their weight have no impact on vehicle targets;
- their efficiency depends on leakage and their resolution.

4.5 E/E Architecture - overview

At this point, the considered E/E components can be grouped in a unique ibd (see Figure 4.23) to have an overview of the defined E/E interfaces between components.

Obviously this refer only to the considered example. When all system components interfaces are defined it become very large due to the big amount of information to be shared and several ibd can be used to focus to the interfaces of a certain component, for example all the interfaces with vehicle components and HCU. Component functionalities, interfaces and consideration are used as input to signal definition and physical layers specification (see Chapter 6).

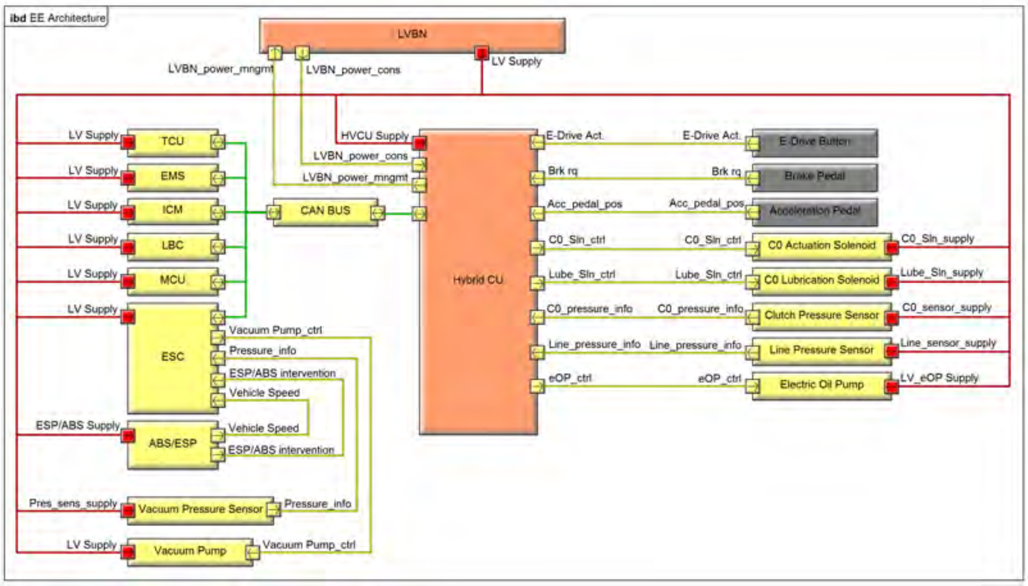

Figure 4.23: Internal block diagram representing E/E interfaces

5 Methodology - Integration

Integration is applied to every level in order to prove that the choices done allow to fulfil RQs and targets. Then this phase is very important to understand if an element or component are selected in the right way and is important apply integration at every level to avoid error that otherwise would be cascaded to lower levels. In fact, if an error is done at level 2 and then wrong RQs are decomposed to level 3 and level 4, these have to be corrected or changed and the effects would be visible at all the three levels. At the same time comport a clear waste of time and money and increase the cost of the project. So that integration is actuated using simulation tasks (both software and testbed). Due to the different levels in the process and the different type of elements and components and depending also on what parameters have to be analysed and simulated, different simulation tools are necessary for this purpose. For instance for vehicle simulation and component at level 1 or 2, AVL CRUISE is suitable but for components at lower level or other components not used by CRUISE (as E/E components), MATLAB®/Simulink or other tools have to be used. However, integration means also communication/discussion between teams of the same department, between teams of different departments and with the customer. In fact, component evaluation has not only to be done looking at component RQs like weight, cost, operating temperature and operating voltage but also looking to customer wants and the needed of the other teams involved because a component selection has effect on both component own level and other levels.

5.1 Examples of integration phases

As already said above, different simulation tasks are applied during integration, depending on the level, the system, element or component that has to be simulated and on the parameters of interest. Example of integration step on level 1/level 0 is vehicle simulation. Usually it is done using CRUISE alone or integrated with other tools like MATLAB®/Simulink, CAR Maker etc. Other example of integration at this level is the first study of PT development done with an approximative elements/components sizing, which can be done with an MS Excel worksheet or a MATLAB®script. At the same time, also vehicle on testbed is an integration step from level 1 to level 0, while PT test on testbed means system integration (from level 2 to level 1). Component integration (from level 3 to level 2) can be both software simulation and physical simulation. For instance, clutch operations can be tested both by a hydraulic circuit solver/simulator (or also Simulink) and by a hydraulic test in laboratory. These tests can be for instance: durability validation, component/system thermal management, mechanical and thermal stresses evaluation, failure condition, etc. For what concern CAN network, CUs, and SW integration, it can be done both with software like Simulink and physical simulation using special electric and electronic embedded systems.

Following, the example used to explain RQs cascading, are taken under consideration to show specifically integration phases at different levels.

5.1.1 Vacuum pump

After component evaluation and selection, and with the parameters from supplier, a simulation is necessary to start the integration phase and prove that component match RQs. In this case, vacuum pump shall maintain vacuum level inside the defined pressure range under several conditions. To do this Simulink can be used, but some information are missing like leakage characteristic of vacuum brake booster, the influence of brake pedal pressure on leakage, vacuum brake booster volume. With this data would be possible calculate energy/power consumption during driving cycle and define critical point for different brake conditions. Other inputs for simulation, already available, are:

- vacuum-time curve (of vacuum pump) but is not referred to vacuum brake volume;
- needed brake torque/power/energy from cycle (available from CRUISE simulation);
- vacuum brake booster force multiplication curve;
- vacuum pump nominal voltage;
- vacuum pump max current;
- absolute pressure range of vacuum brake booster.

Vacuum pump Simulink model allows to:

- estimate power consumption;
- estimate average power consumption in the studied time range;
- estimate the behavior correspondents to on-off pressure range;
- estimate pump average suction capacity in the considered pressure range;
- estimate if the chosen pump fulfills RQs.

Notes:

1. Relationship between leakage and pressure and between driver braking force and leakage are missing (very important to correctly simulate pump behavior);
2. Switch on pressure usually depends also from altitude and the pressure range for hysteresis control depends on control loop.

In this case study, the pressure range is 500-300 mbar (absolute pressure) and the vacuum pump runs for three seconds to pass from 500 mbar to 300 mbar. The characteristic/behavior simulated are visible in Figure 5.1.

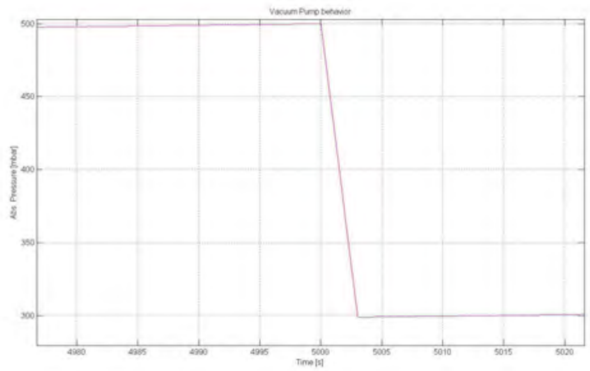

Figure 5.1: Vacuum pump behavior (Simulink)

This simulation is done at level 3 (level of vacuum pump) to integrate component in level 2 (brake system).

5.1.2 Separation clutch

After separation clutch characteristics are defined, before to go to level below and evaluate other components leading to it, is important verify that it is able to fulfil the RQs. Several test/simulation can be done, for example durability, validation of torque-pressure curve,thermal behavior, etc. In this case the effect of closing time in vehicle fuel consumption is studied and CRUISE is used. The results are summarized on Table 5.1. Consideration:

1. during NEDC and WLTC there are no appreciable variation of FC (clutch is closed few times);
2. see the effects on performance have no sense because clutch is always closed;
3. so that, a real life driving cycle (RLDC) is considered.[1]

Table 5.1: Effect of clutch closing time on vehicle fuel consumption

Closing time	**Fuel consumption**	
[s]	[l/100km]	%
0.05	5.899	(-0.25)
0.2	*5.914*	*Reference*
1	5.92	(+0.10)
2	5.909	Error!

Note that the fuel consumption in the last row is lower than the reference value. So that simulation results cannot be considered because the real fuel economy is lower than simulation accuracy (numerical errors). Then other tools are necessary to quantify the effect of closing time. This simulation corresponds to the integration step between level 2 and level 1.

5.1.3 Separation clutch control system

With components attributes, different simulations can be done to prove RQs fulfillment. In this section two examples of them are shown. The first regards to P2 module components LV energy/power consumption during two different driving cycles, the second shows a virtual hydraulic simulation of the clutch control system. These simulations represent integration step from level 3 to level 2.

5.1.4 P2 Solenoids and electric oil pump energy/power consumption

In this case, knowing components data and their control method, a simulation in Simulink is done to estimate their energy consumption during NEDC and WLTC, which speed profiles are reported respectively on Figure 5.2 and Figure 5.3. This energy then can be used to calculate the total energy/power consumption of the low voltage board net. In Figure 5.4 and Figure 5.6 the results of simulations are shown. The peaks of P2 module LV power consumption corresponding to the engine start. The times in which the engine shall be started depends on the driving cycle and the HV battery state of charge. In fact for both NEDC and WLTC, the final SOC of the HV battery shall be equal to the initial SOC with a margin of $\pm$ 0.5% of SOC.

From simulations, results:

- during NEDC energy consumption is 29.77 Wh;
- during WLTC energy consumption is 50.59 Wh.

[1] There are several RLDC, composed by different combination of urban and highway tracks and slope grades. They are used to simulate vehicle behavior in a driving cycle more closed to the reality.

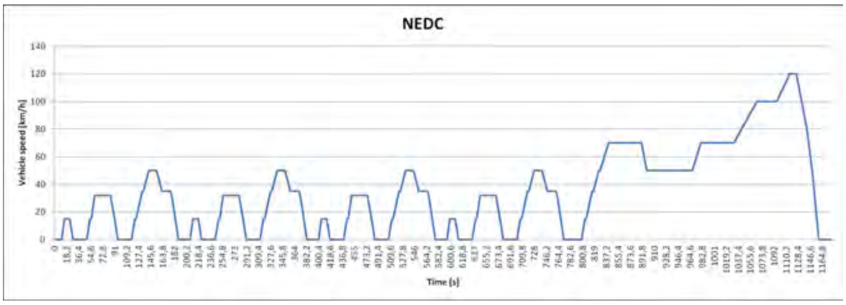

Figure 5.2: New European Driving Cycle - Speed profile

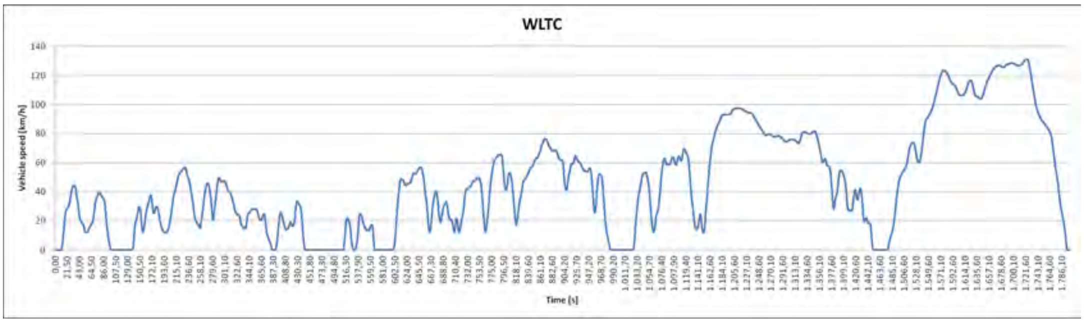

Figure 5.3: Worldwide harmonized Light duty driving Test Cycle - Speed profile

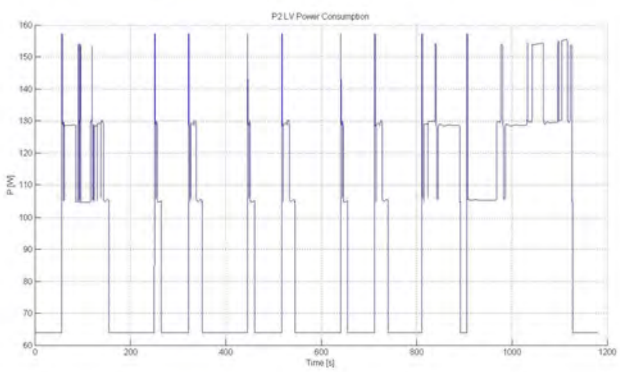

Figure 5.4: NEDC P2 LV power consumption

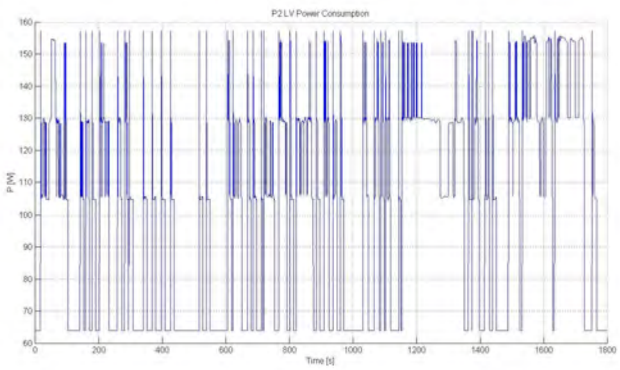

Figure 5.5: NEDC P2 LV power consumption

5.1.5 Hydraulic circuit simulation

Hydraulic simulation is necessary to study clutch behavior with the selected components (solenoids, pressure sensors, electric oil pump). Usually hydraulic simulators are used for this purpose, but also Simulink can be used. To implement the model, all components behavior shall be known and several assumption are needed. Luckily, in Simulink library there are several toolboxes with different components, and *"Simscape"* folder is suitable for our application. It contains mechanical, hydraulic and electrical components and other utilities. Nevertheless, at the same time these components required to know a big amount of information, which usually are available after components selection. Is very important to note that the initial effort to model hydraulic circuit is then reward for the possibility to integrate the circuit with components control signals always built in Simulink environment. After built the hydraulic model (see Figure 5.6), the clutch closing phase is simulated controlling both the solenoids with a step signal. The results reported on Figure 5.7 shows that closing time to kiss point is about 130 ms. Due to less information available for components and the not applied control strategy to oil pump, there are some imprecisions in the results.

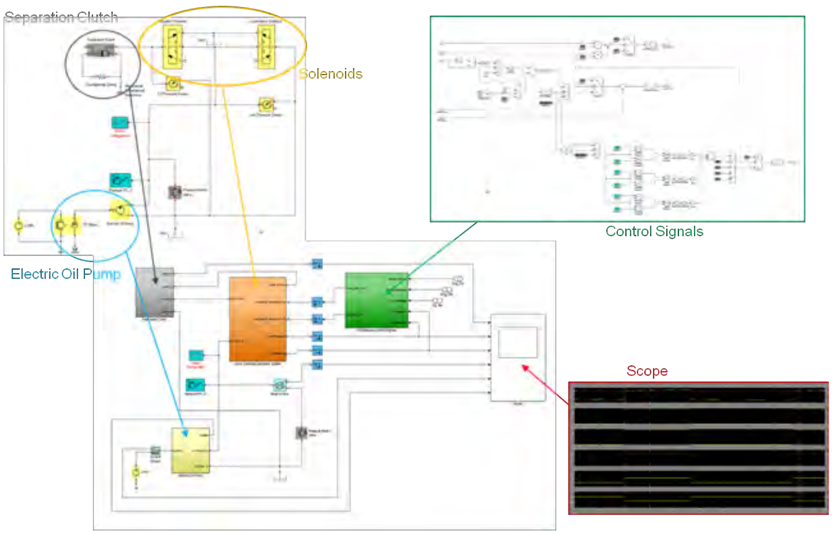

Figure 5.6: Simulink model of hydraulic circuit

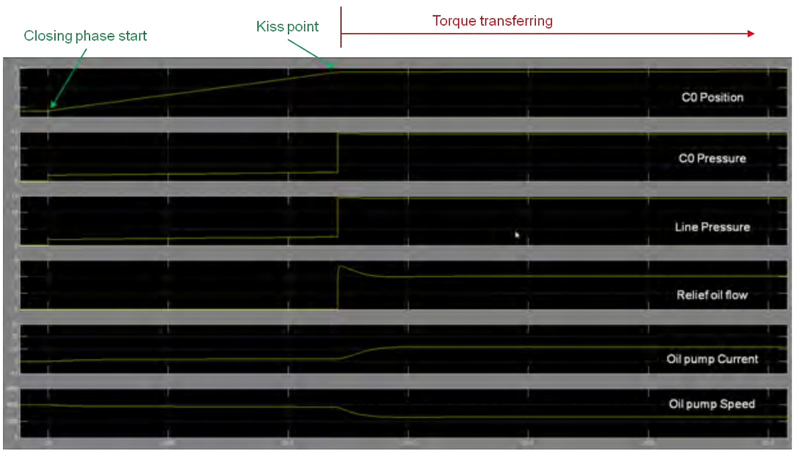

Figure 5.7: Simulation results

5.1.6 Vehicle simulations

Usually these simulations are done in CRUISE and sometime Simulink is integrate to better implement the control of vehicle elements/components. However, in the early phase of a project, when few information are available and the powertrain topology investigation has to be done in few days, CRUISE become unusable. In fact, build a CRUISE model need more time and a lot of information. Then, other tool like the Excel worksheet for fuel consumption and performance calculation actually used in the first phase of a project, or a MATLAB®script with the same purpose are more useful.

First phase simulation

Following is described the first phase simulation when not all the components are defined and less information are available. Vehicle target, that then become a RQ, fuel consumption reduced of 28% than conventional vehicle, is taken under consideration. Looking at the consideration done in section 4.2.1. and also looking at Figure 7, the RQ can be fulfilled using different solution, for example improve PT total efficiency (injection system control, transmission control efficiency, etc.) or improve vehicle attributes like weight, rolling resistance, drag resistance, frontal area, etc. The improvement "powertrain hybridization with P2 configuration" is chosen but that cause an increment of weight (about 100 kg due to E-Motor, HV Battery, separation element) and an increment of LVBN power consumption (supposed 100 W). Now, with the hybrid model with also improvement on engine and transmission (Atkinson cycle, transmission control strategy) the simulated fuel consumption is equal to 23.8% less than conventional vehicle. From this, become the conclusion that with only PT hybridization is not possible achieve the target. Other improvement shall be done or the PT hybrid topology shall be changed. Assuming to improve the rolling resistance (chassis attribute), a new simulation shows a fuel consumption reduction of 28.2% and in this case RQ is fulfilled. Note: usually this first step, that can be considered the first integration step, is done not with CRUISE but with tools mentioned above.

Vehicle model with CRUISE and Simulink integrated

Another example of integration step to demonstrate that all the elements/components selected allows to achieve vehicle targets, is the following. The CRUISE vehicle simulation model has been modified adding/integrate the Simulink model of E/E components to simulate with a good accuracy the power consumption during the driving cycle. This tools integration can also be used to calculate maximum, minimum and average electric power consumption and the energy power consumption and this can be useful for control strategy definition and HV Battery behaviour study. With the following considerations, several simulation tasks are done and the results are visible on Table 5.2 and Table 5.3. Data used for CRUISE simulation tasks:

- NEDC: HV Battery SOC 59%, ICE minimum ON time 18 s;
- WLTC: HV Battery SOC 55%, ICE minimum ON time 18 s, A/C OFF.

Table 5.2: Vehicle simulation results

		Fuel consumption		**Performances**	
Type	Curb Weight	NEDC	WLTC	0-100 km/h	80-120 km/h
[-]	[kg]	[l/100km]	[l/100km]	[s]	[s]
Conventional Vehicle	1595	7,770	7,732	8,86	6,38
Hybrid Vehicle	1700	5,623	6,059	9,07	6,16
	(+6,58%)	(-27,63%)	(-21,64%)	(+0,21 s)	(-0,22 s)

In Figure 5.8 an example of AVL CRUISE window containing vehicle elements and components.

Table 5.3: E/E components power and energy consumption

	NEDC	WLTC
PMAX [W]	700,21	962,16
Pmin [W]	362,26	439,31
Pavg [W]	425,38	548,84
Cycle Energy Cons. [Wh]	139,41	274,39

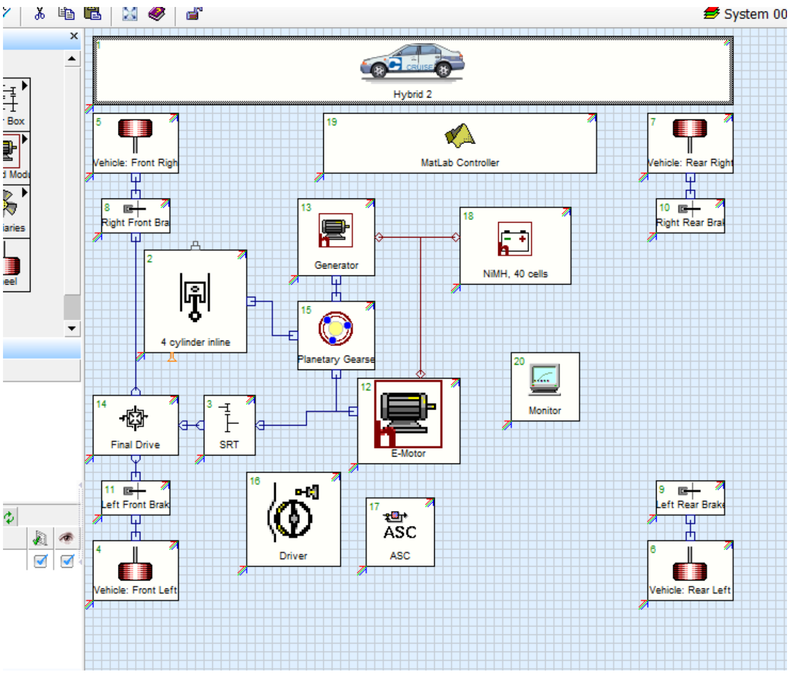

Figure 5.8: AVL CRUISE main window

6 Methodology - Next steps in E/E development process

6.1 Physical layers specification

After RQs cascading and integration steps to prove that selected components allows to achieve the targets, regarding only E/E the next phase is that to define physical layers (this means also interfaces specification). This phase is possible only after that a component (not only CUs) is selected. From component data (provided from component's supplier) and control method, is possible to determine number of wires, wires attributes (cross section, insulation, etc.), type of signal, type and attributes of connectors, CAN interface, etc. Furthermore, other E/E RQs can be derived like:

- maximum allowable drop voltage shall be 0.1 V (usually maximum permissible drop voltage for automotive device is 0.2 V and for computer (CUs) is less than 0.1 V);
- cable shall be protected by fuse (usually is done when the device is a high power consumer and the current due to device failure or short-circuit can damage the cable);
- wires harness shall have working temperature range comparable to vehicle module working temperature range plus a certain heat amount due to current flow;
- wires shall be shielded;
- cable mechanical RQs (e.g. resistance to traction and to torsion).

For CAN messages definition, the inputs are source of message, sink (or sinks if more devices need the information sent) and the content of the message. The following examples is based on the E/E components analyzed on the sections above. Starting from their data, their E/E interfaces, their control method and the considerations done, their physical layers can be specified. Due to the nature of components, no CAN messages are specified but in case of usage of devices with CAN interface, the physical layer specification would refer to CAN messages specification.

6.1.1 Example - Vacuum pump

In this case study, vacuum pump is only turn on and off and then does not need a control signal but only a supply wires with a switch controlled by some CU (tbd) which elaborates pressure sensor information. In order to define supply wires attributes, several input info are needed:

- maximum current;
- maximum temperature (for insulation), this can be derived from temperature range RQs listed in level 0 (vehicle) and depends on the location of the vacuum pump respect the PT sources of heat (e.g. engine, exhaust system);
- nominal voltage of LVBN (in this case 14 V);

- wires conductive material (copper, aluminum);
- wires length (an estimation if not available) that depends on component location;
- mechanical RQs of cable.

6.1.2 Example - P2 module pressure sensors

As for the vacuum pump, starting for the considerations done in Section 4.4.4 is possible specified the physical layer of P2 pressure sensors. From its building characteristic (three terminals), derive the necessity to use three cables: two for power supply, positive and ground, (supply interfaces) and one for output signal (information interface). Then, the cross section can be 0.5 mm^2 due to the low current but with some additional measures (regarding overall to wire mechanical resistance) also a cross section of 0.35 mm^2 is permissible. The same consideration can be done for vacuum brake booster pressure sensor.

6.1.3 Example - P2 module solenoids

From solenoids consideration and main data, is possible to say that:

- solenoids control interface consists of a wire that transmit the calculated current in order to provide a certain oil flow (and then increase/decrease pressure). Its cross section depends on max current and wire length;
- solenoids supply interface consist in only one cable that corresponds to ground.

6.1.4 Example - Electric oil pump

After pump is selected from the preliminary design and the considerations, also a control method shall be defined. Different types of control can be actuated but a step control is used due to the low response of the pump to control variations. Otherwise, other control strategy shall be analyzed like hysteresis control. With its data is possible better define interfaces attributes:

- two wire are needed for supply interface and their cross section depends on max current, length of the wires and allowable voltage drop (assumed 0.9 V);
- one wire is necessary for control signal (control interface) and its section depends on control signal characteristics.

Is important to observe that electric oil pump is a high power consumer and that affect LVBN power consumption and then vehicle RQs and targets. Its evaluation shall be done with criteria. Furthermore, due to high current in case of component failure, cable shall be protected by fuse (new E/E RQ).

6.1.5 Summary

After cross section calculation, done assuming wire length and using components data and looking at cable RQs, the cables are chosen from supplier's products. The standard cross section chosen for the analyzed components, are summarized in Table 6.1.

Table 6.1: Wires max current, cross section and fuse value

Component	Max current [A]	Cross section [mm^2]	Fuse [A]
Electric oil pump	~22	10 (AWG 8)	30
Vacuum pump	11	4 (AWG 12)	20
Solenoids	1.3 - 1.5	0.5 (AWG 21)	-
Pressure sensors	Tens of mA	0.5 (AWG 21)	-

6.2 Signal and CAN-Matrix definition

After control method selection, the signal to be used to control is defined with its attributes like range and resolution. Then, during physical layer specification phase, in case of CAN signal also its cycle time is defined with the main information (source, sink and description of the message). Finally, signals are assigned to CAN message. This results on the so-called CAN-Matrix, a table with a list of all CAN messages and signal layout. Software engineers then use CAN-Matrix as input to develop control units software. This step has not been applied to the considered components.

6.3 General consideration

Applying the methodology, several issues and aspects can be found, and some considerations are summarized in this section.

- Better features and functions definition leads to better interfaces definition (more details at higher levels). In fact if features and functions are very detailed and well known at the start of the development process, inputs and outputs are easy defined and with them also the interfaces between systems, elements and components;
- Another important consideration is regarding system structure (what components at what level). In fact, if it or a large part of it is specified at the initial stage of the development process, architectures can be quickly derived as like RQs;
- From interface definition become the conclusion that there are 5 main E/E interfaces: LV supply (from LVBN to other E/E components), HV supply, HMI (usually from body to other components), information (shared from components), control. Currently the first two are grouped and modeled as electrical energy interfaces and the other are grouped and modeled as information interfaces;
- In the most general case, on level 3 (components) there are only supply, information and control interfaces (e.g. separation clutch, E-Motor) and on level 4 (sub-components) only supply and control interfaces (e.g. solenoids) or supply and information interfaces (e.g. pressure sensors);
- Going through levels, new components can be found and new interfaces defined, so that boundary diagrams and interfaces shall be updated and in particular for interfaces, more information can be added (e.g. clutch control interface can be separated in actuation solenoid control and lubrication solenoid control). At the same time, also new signals/messages can be defined and both the model and the physical layers and messages list need to be continuously updated;
- Sensors that provide information that shall be shared, can connect to the own CU with a hard wire and then their info are shared via CAN (or other network) with the other CUs or can be direct connected to the CAN network (this solution increase the cost but increase information sharing efficiency);
- E/E interfaces are derived from RQs and features/functions, and are updated after lower level RQs and architecture are defined;
- E/E interfaces shall be: general at the highest level and/or when no much information are available (a component shall share some info and shall be controlled but we do not know how and from who); more detailed as possible when more information are available (component A shall share voltage level with components B and C, and it is current-controlled by component D);
- E/E RQs can be mechanical, electrical and thermal:
  - Thermal RQs refer to operating and storage temperature range;

  - Mechanical RQs refer to vibrations and stresses (include also resistance to traction, torsion and chemical agents);
  - Electrical RQs refer to operating voltage, connection types, insulation, cable protection, components involved on communication, etc.

- Thermal and mechanical RQs and some electrical RQs (operating voltage, safety regulations, EMC) are valid for all component, so they can be listed at vehicle level and then used as boundary condition for component evaluation;
- The rest of electrical RQs (cable protection, components involved on communication, etc.) are located on the level in which the related component is;
- Physical layers and messages specification are possible only after that a component and its control method are selected.

7 E/E modeling using SysML™

Current AVL approach is based on MBSE. The benefits deriving from the transition from document-based approach to model-based approach, are reported in Section 2.1. The advantages of all the information included in a model is clear. The tool used for system modeling is Artisan Studio SysML™, general information about it can be found in Chapter 2.2 or in Bibliography.Of course there are different methods to model a system using a specific tool like SysML™ and its diagrams and several personalization can be applied like colors, levels organizations, etc. This chapter is focused only on physical layers representation and integration in system model, included wire attributes using AVL stereotypes. At the same time the possible representation of E/E architecture are discussed.

7.1 E/E views

As already said, have no sense to use different diagrams (ibd) for every E/E component and its interfaces or physical layers, but usually the integral E/E architecture is represented in only one diagram (see Figure 7.1 as an example).

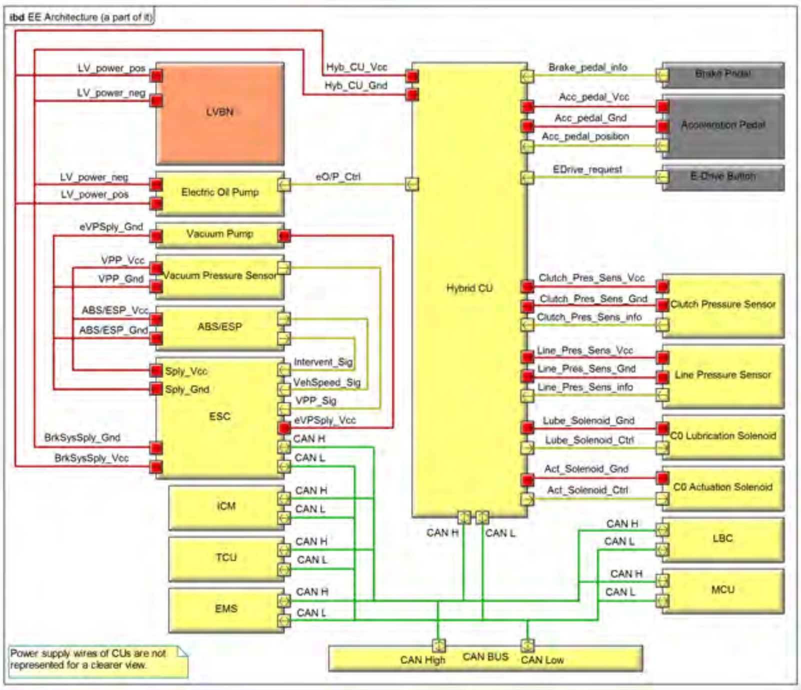

Figure 7.1: E/E architecture with CAN network

The diagram do not include all vehicle E/E components but is important to note that due to the high number of components and wires, the diagram could be very confusing. So that, instead of only one diagram, for a clearer view several ibd can be created, focused only to certain components or certain physical layers really needed for a determined project (see Figure 7.2).

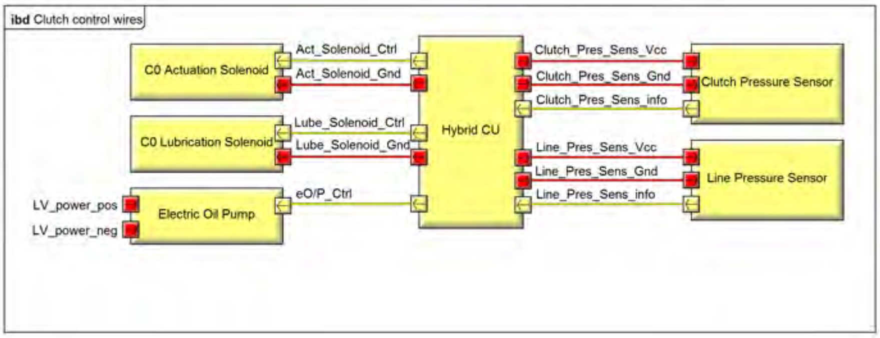

Figure 7.2: Clutch control components and their hard-wired connections

In the system model then, several ibd for E/E views can be done. To reduce the effort, they shall be discussed at the start of every project depending on their purpose and the needed of customer, software and component engineers, electrical engineers, etc. At the same time, the ibd considered necessary for most of the projects, can be included in the template system model to reduce again the modeling effort. Other examples of useful E/E ibd could be:

- all CAN connections (see Figure 7.3);
- signals between a CU and the other components (useful for SW development);
- wires between components (useful for electrical engineers);
- E/E interfaces between components (useful to define physical layers and CAN signals).

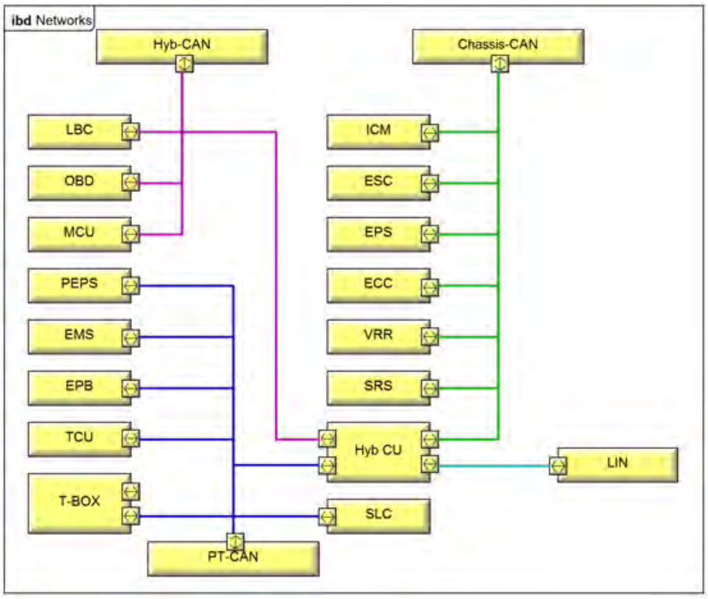

Figure 7.3: CAN connections overview

An important observation: if several E/E views are needed, is better to create a big ibd with all the physical layers represented (included CAN hard-wired connections) and use it as a background. Then from it with simple steps (called "populate" in SysML^{TM}) is possible recall only the blocks, the flow ports and the connectors needed, depending on the diagram desired. Otherwise, if only a pair of E/E ibd are needed, take less effort model them separately.

7.2 Physical layers modeling

Currently, physical layers specification is done as described in Section 6.1. Then they are listed in an Excel file. This file contains the name of the interfaces, type of physical layer, the CAN message name, physical layer attributes (hard-wired attributes and CAN message attributes), the source and the sink of the interfaces and the signal definition. Physical layers can be modeled in the SysML™ system model.

The actual physical layer modeling phase, regards only hard-wired interfaces and can be described with the following basic steps:

1. Both in the source and in the sink block, are added the flow ports (information or electric energy depending on the type of interface);
2. Then these are renamed with the physical layer name;
3. Are connected together with a connector;
4. To the connector both the stereotype (information, LV or HV connector, CAN) and the name of physical layer are added;
5. Finally, all the names are hided without the name of only one flow ports.

The improvement proposed for physical layer modeling and in completely agreement with MBSE approach, consists on adding the following steps:

i At the step 2 of the current approach, over the physical layer name, also wire information like pin assignment, color, cross section, etc. are added to flow ports and in particular are written on "description" window of the properties pane (see Figure 7.4);

ii At the step 5, instead of maintain the name of one flow port, the only name that is not hided is the name of the connector (equal to the name of the flow ports) because it can be moved along the connector. This is helpful for an aesthetic aspect: allows making clearer and ordered the diagram.

iii A new step then, is that to insert the list of CAN messages (Excel table) in the model (see Figure 7.5). The table is linked to the block CAN in the model (to block's description in properties pane) using the option "insert OLE object". This option allows also linking permanently the table; this means that when the file is updated, also the table in the model is updated.

7.2.1 Outlook

In order to complete the passage from document-based approach to MBSE approach, the next step is that to derive an Excel table with all the interfaces and their attributes like the currently used Excel worksheet and if it is possible do the same also for the CAN messages list. Looking at Figure 7.4, a word table for hard-wired physical layers can be derived from the model using Artisan Publisher™.

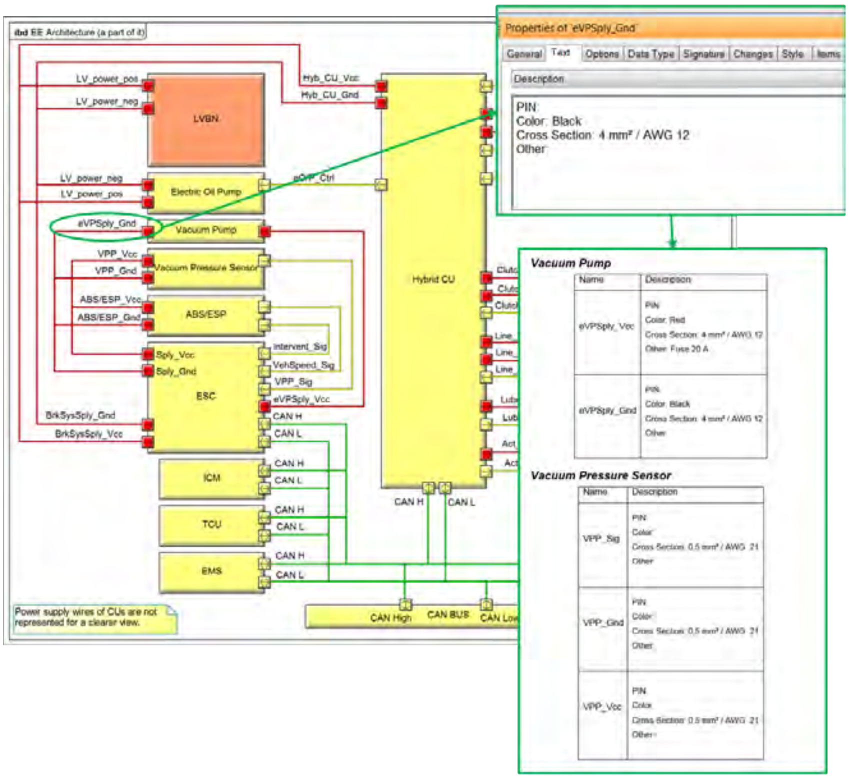

Figure 7.4: Hard-wired attributes representation

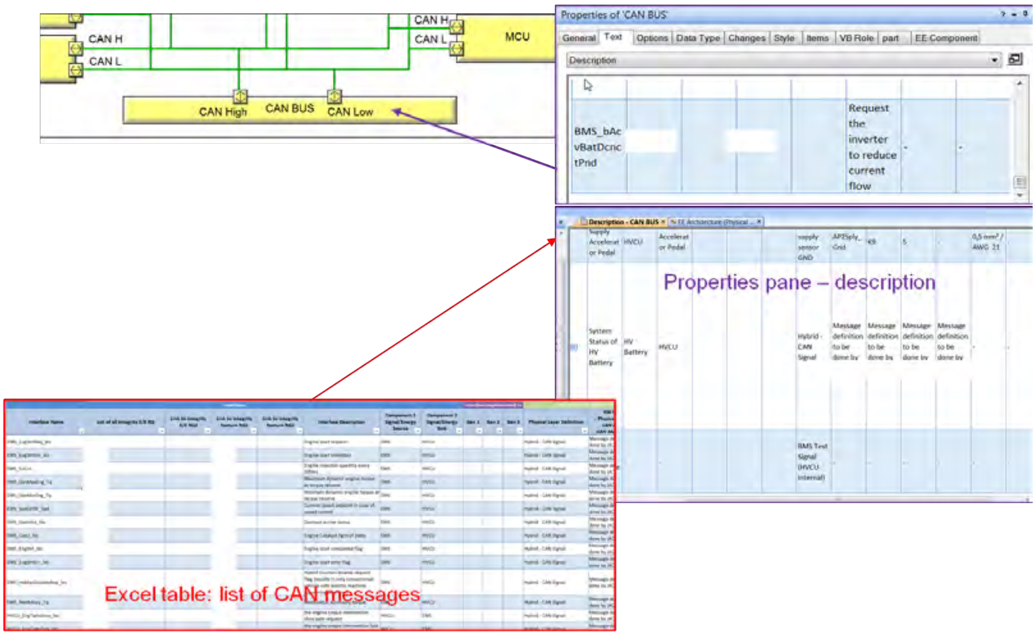

Figure 7.5: CAN messages adding phase

8 System simulation

A chapter is reserved to system simulation due to the major role that simulation tasks have in integration phase. As reported in Chapter 5 different kind of simulation are used depending on the level and the parameters to be verified. Virtual simulations (also called SW simulations) increasingly substitute physical simulations (e.g. testbed) and allow reducing development costs. Then is important to plan simulations and modeling together and this has to be done early in the development process to save money and time after. Actually, in the AVL development process, this link is missing and results in different interpretations of the information available and no communication between system engineers and simulation engineers. In order to optimize the development process, a parallel way to the methodology described in this report regarding system simulation, shall be defined.

8.1 Actual simulation tasks

Actually, system simulation and system modeling work on two different and separate way. System modeling collects all the information from "SharePoint" but not all them (for example element/component maps and curves) and this because the passage from document-based approach to MBSE approach has not yet been completed. At the same time, system simulation use more element/component specific information and the only information required to system modeling are related to vehicle and systems features, and the allowed transitions between them. This results in a big gap between these two environments, with clear effects on development process. Another aspect to be considered is that simulation tasks are not used at every level, this means that the integration phase is not completed and again this badly affects the development process (in costs and time). Finally, the fact that at the moment there are not direct links (at SW level) between the main used tools for system modeling and system simulation (i.e. SysML and CRUISE), is for sure not helpful to fill the gap.

8.2 First steps to link simulation and modeling environments

During the application of the described methodology to E/E components, the needed to prove that the selected components allow fulfilling RQs (integration) drive towards simulation environment. As already demonstrated, at components level CRUISE cannot be used, whereas Simulink is very suitable and then can be integrated with CRUISE. For what mentioned above, find a link between SysML and Simulink or SysML and CRUISE, would be very useful. Actually, there is not a direct link between SysML™ and CRUISE, and MATLAB®can be used like an interface between them (included other tools like MS Excel). As previously said, find a direct link would be very important for information sharing and time saving. Regarding E/E components, Simulink is more useful for component simulation and control method validation and luckily there is a direct link between SysML™ and Simulink. However, the information that are possible to share are limited. In order to import/export information from/to Simulink, Simulink Profile has to be added to the model. With it is possible import/export internal block diagrams, parametric diagrams and constrains, classes (not used) and RQs (RQs can be only imported from Simulink to SysML™). More information can be found in Appendix A.

Currently, the most used item that can be export/import, is the ibd. Following an example of its exportation and modeling in Simulink.

8.2.1 Example - Electric oil pump

In this example, is shown a possible link between tools. Obviously is very simplified and at the same time is not completely correct because the inputs shown are not the real input for electric oil pump but would be the inputs for its control. In Figure 8.1, the procedure is shown and the main steps could be summarized as follow:

1. build in SysML™ the ibd that has to be exported in Simulink (to do this, several rules shall be followed: see Appendix A or Artisan Studio® "Help topics");
2. export the items using Simulink Synchronizer specifying how they will be shown in Simulink model;
3. now only the "shell" of the model is imported in Simulink (blocks as subsystems/model reference, flow ports as inputs/outputs, connectors as lines) and is possible to model the components with Simulink blocks;
4. finally, the model can be simulated in Simulink or integrated in CRUISE and simulated.

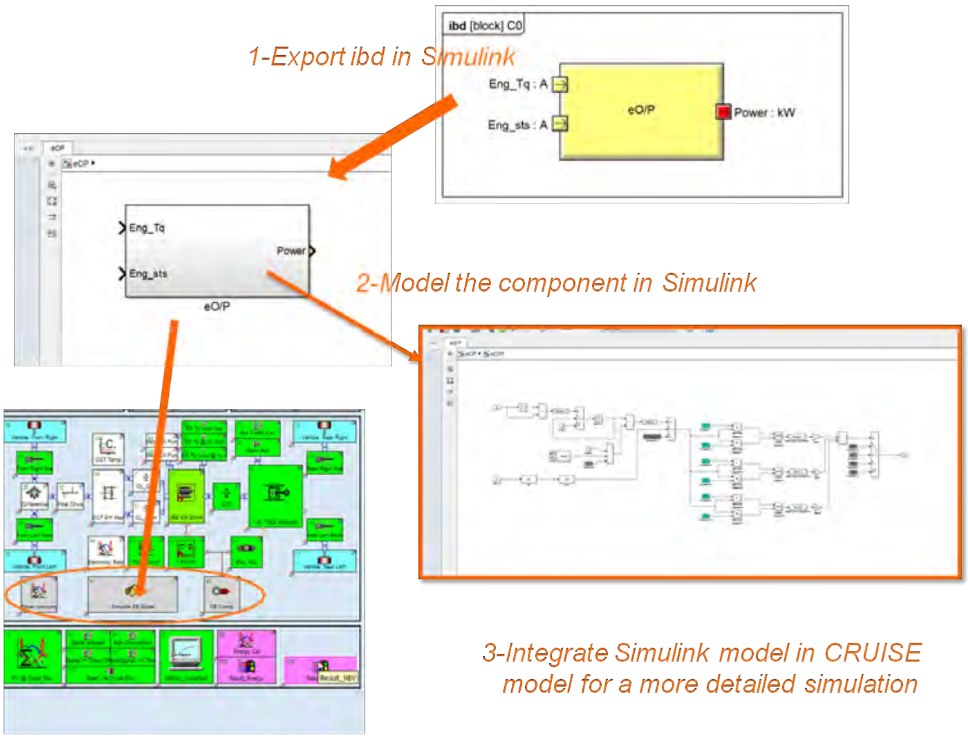

Figure 8.1: Tools integration

Note: this example help to understand the importance to have links between different tools to improve results and reduce modeling effort.

9 Conclusion & Outlook

During the methodology application, several issues are found and from these some important conclusions are derived. Those mainly refer to feature/function definition, system structure definition, integration and department's communication. To correctly apply the methodology and optimize the working time, at the initial stage of the development process features shall be well defined. The same is valid for the system structure. In fact, well-defined features (input/output, list of sub-features, transitions) and the knowledge about what component is in a determined level, lead to a well-specified architecture and, therefore, the right amount of useful information is available at higher levels. This means that interfaces and RQs are easily found and the working time of these phases is reduced. In the traditional approach, the alignment phase is limited and, as reported in Chapter 5, a good planning of such an alignment would allow saving money and reducing development time. In fact, such an alignment or integration phase would be applied to every step and would be immediately clear if a component evaluation has been done correctly. To plan integration means define a simulation path (methodology and workflow) parallel to the E/E development process with several links between them. Finally, the current development process is badly affects by missing communication between teams as well as from the non-completed transition from document-based approach to MBSE approach. In fact several times can be noticed that different teams have different information but is very difficult to define what information are the best and at the same time on SharePoint there are a lot of documents that could not be updated and that contain data in conflict with them. From these conclusions, other improvements can be analyzed and applied in the future. Some of these are related to department's communication, to complete the transition to MBSE (with the benefit to have all the information in one model and these are always the best, i.e. always updated) and to define and plan a simulation path to provide and improve integration process.

Information about possible future improvements regarding E/E modeling and aligned with MBSE approach, are described in Section 7.2.1.

For what concern system simulation, as reported in Section 8.1, in order to reduce effort, costs and time, is necessary firstly to define a joint plan, formed by two parallel paths linked when it is needed. One of this is the methodology described and the other is a new methodology for system simulation (to be defined and developed). The initial high effort, cost and time to develop this new path, then will be reward to all the benefits that a joint and aligned work comports. The next step, after the definition of this new "simulation path", is that to define when the two ways shall be joint, and for instance the junctions could be the integration steps through the levels. Finally, in order to totally fill the gap, the passage to MBSE approach shall be completed. Then, all the information are allocated to the model (they shall be always updated) and are used from all the teams. This means that all the information will be unique and no mistake will occur. The information to be included shall be discuss with all the teams in order to use only the needed information and data and to avoid high effort for modeling, long working time and a huge amount of information that could make confusion. Another important improvement to do, in order to align simulation and modeling environments, is that to link (from a software point of view) SysML and the simulation tools (CRUISE, Simulink, etc.). It would be very helpful for a correct information sharing and at the same time would allow saving money and again reducing development time.

A How to exchange data between SysML^{TM} and MATLAB$^{®}$Simulink

In this section, is explained how to pass from Artisan Studio SysML^{TM} to MATLAB$^{®}$Simulink and the opposite, using the "Simulink Profile" in SysML^{TM}. More details can be found in Artisan Studio$^{®}$"Help topics" in the program. First, before importing/exporting items, is necessary to verify that the installed version of SysML^{TM} is compatible with the installed version of MATLAB$^{®}$/Simulink . Then, the "Simulink Profile" shall be added in the SysML^{TM} model. To do this, open **Tools** menu in SysML^{TM} and select **Add profile**, select **Simulink** entry and click **ok**. After the profile installation, is advised to update all the model profiles (SysML^{TM}, UML^{TM}, Simulink, etc.).

The SysML items that are possible to import/export are:

- Parametric diagram;
- Internal block diagram;
- Class;
- Requirement (can be only imported from Simulink).

In Table A.1, are listed the complementary Simulink items of SysML^{TM} items.

Table A.1: SysML^{TM} items and relative Simulink items

SysML^{TM} Item	**Simulink Item**
Internal Block Diagram	Simulink Model
Block and Block Property	Model Reference/Subsystem
Atomic Flowport	Inport/Outport
Connector	Line
Parametric Diagram	Simulink Mode
Constraint Property	Model Reference/Subsystem
Constraint Parameter	Inport/Outport
Value Property	Data Store (Read/Write)
Connector	Line

Following, some examples and important steps about import/export of parametric diagram, ibd, class and RQ are listed. Regarding parametric diagrams, before to export them in Simulink, to ensure that constraint parameters are exported to a Simulink model, is necessary to create (if it has not be done) a child parametric diagram for every constraint block and then populate them with the constraint parameters.

Taking as an example the parametric diagram in Figure A.1, the exportation can be done as follow:

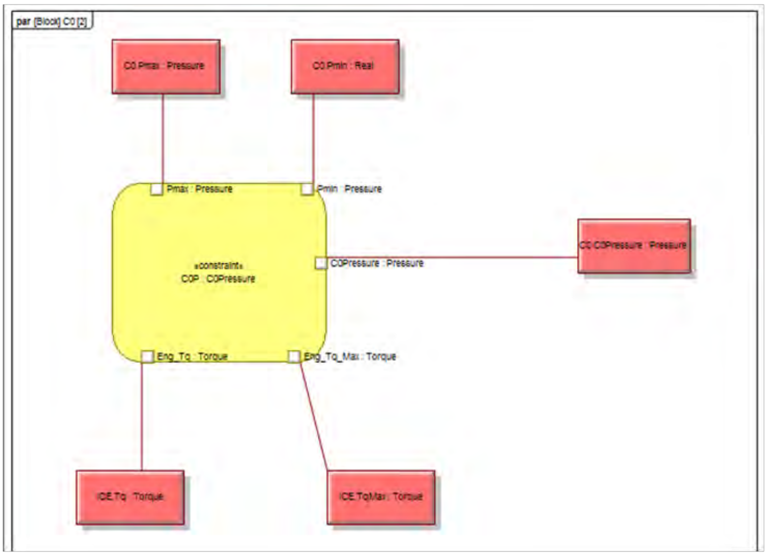

Figure A.1: Parametric diagram to export in Simulink

- In Studio, right-click on parametric diagram (it can be done on package pane or in the opened diagram), point to **Tools**, point to **Simulink Synchronizer** and click **Synchronize**;

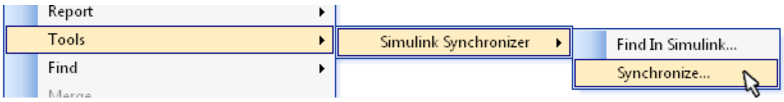

Figure A.2: Exportation - Step 1

- Set the options on each page of the Synchronizer as desired;

Figure A.3: Exportation - Step 2 Create new map

Note: when exportation is complete, control the presence of eventual warnings or errors in the dialog window. A warning or an error can lead to a not complete or a wrong exportation.

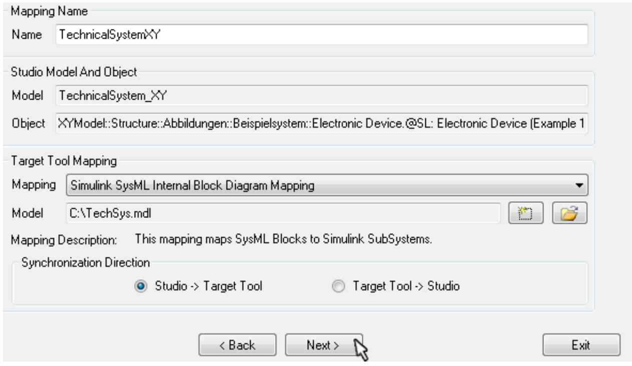

Figure A.4: Exportation - Step 3 Map options

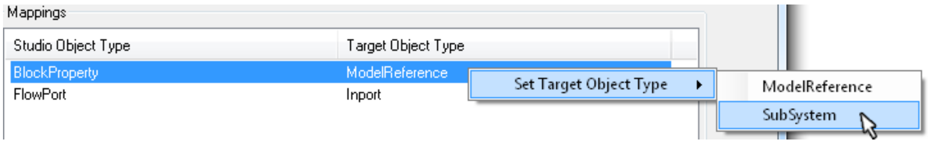

Figure A.5: Exportation - Step 4 Set Target Object

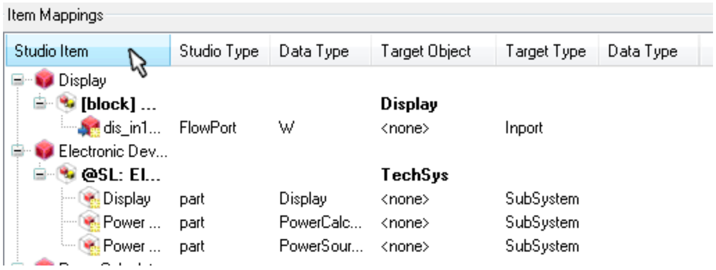

Figure A.6: Exportation - Step 5 Control how the item are exported

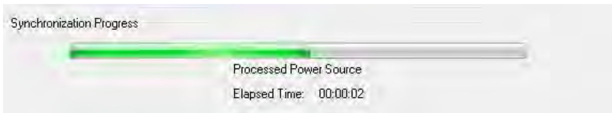

Figure A.7: Exportation - Step 6 Start exportation process

- After the exportation process, the result will be like Figure A.8.

Note: formulas contained in parametric diagrams are not exported in Simulink, however modifying the Visual Basic (VB) scripts of Simulink Profile, also their exportation can be possible.

These steps are independent from the item to be imported/exported and then are valid also for ibds, classes and RQs.

One time that the shell is exported on Simulink, is possible to build up the model in Simulink with the appropriate blocks.

Applying the opposite process (importation), is possible passing from Simulink model to SysML parametric diagram.

- Ensure that the Simulink model with which you are working is not open in Simulink;
- In Studio, right-click the Block or Constraint Block that is to own the Parametric Di-

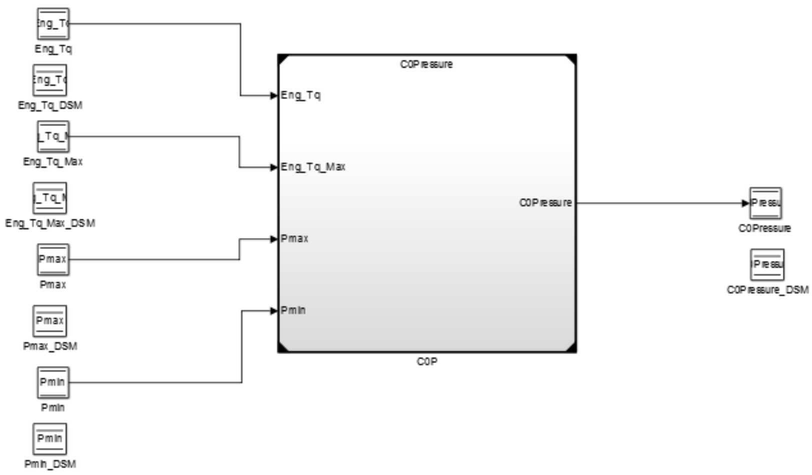

Figure A.8: Simulink model of parametric diagram

agram, point to **Tools**, point to **Simulink Synchronizer**, and then click **New Parametric Diagram**;

- Alternatively, Create an empty Parametric Diagram, right-click the Parametric Diagram, point to **Tools**, point to **Simulink Synchronizer**, and then click **Synchronize**;
- Set the options on each page of the Synchronizer as desired;
- The Synchronizer imports the Simulink model to the Parametric Diagram.

For opening the Simulink block diagram associated with a Studio parametric diagram, in Studio right-click the Parametric Diagram, point to **Tools**, point to **Simulink Synchronizer**, and then click **Find In Simulink**. The Synchronizer opens the appropriate Simulink model and block diagram.

The Synchronize command is used to export a Studio Parametric Diagram, Internal Block Diagram or Class from a Studio model to a Simulink model the first time. Thereafter, the Synchronize command is used to update Simulink objects and diagrams from a Studio model, or to update Studio items and diagrams from a Simulink model.

The second item that is possible export in Simulink, and looking at the purpose of AVL Integration Process also the main one, is the internal block diagram. As for constraint parameters in the parametric diagram, to be sure that the flow ports of an internal block diagram are exported in Simulink, those Flow Ports must appear on an Internal Block Diagram that is owned by their owning Part's type (i.e. a child internal block diagram). So that, if there are no child Internal Block Diagrams for every block, create a child Internal Block Diagram: right-click the Block, point to **New**, and then click **Internal Block Diagram**. On the child Internal Block Diagram, right-click the diagram background, point to **Populate**, and then click **Flow Ports**. Note that in Simulink there are only inport and outport whereas in SysML^{TM} there are also in/out flow ports. For this reason, the flow ports shall be atomic (i.e. they can be only set to in or out). Using the steps shown for parametric diagrams, is possible export/import an internal block diagram, update it or open the Simulink model linked to it. In Figure A.9 an example of ibd exportation and in Figure A.10 an example of ibd importation.

Regarding Classes and RQs, actually these are not used in AVL modeling so that only an example of importation of them are reported (see Figure A.11). Again, the steps to do are the same that are used for parametric diagrams and ibds; again is possible to find other information in Artisan Studio$^{®}$"Help Topics".

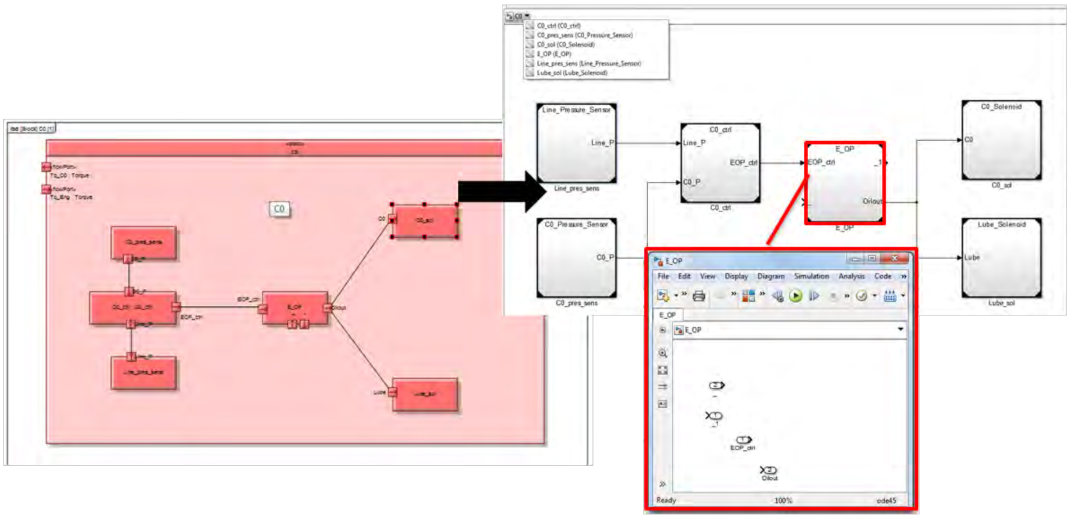

Figure A.9: Ibd export

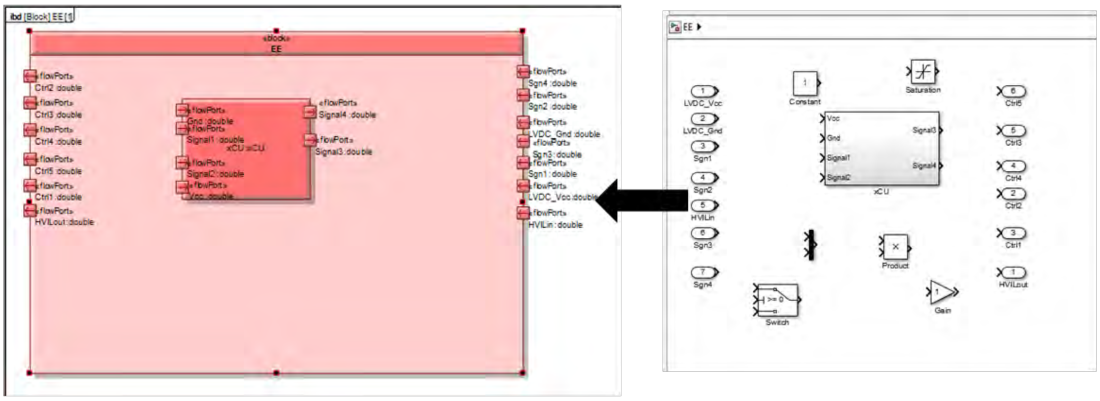

Figure A.10: Ibd import

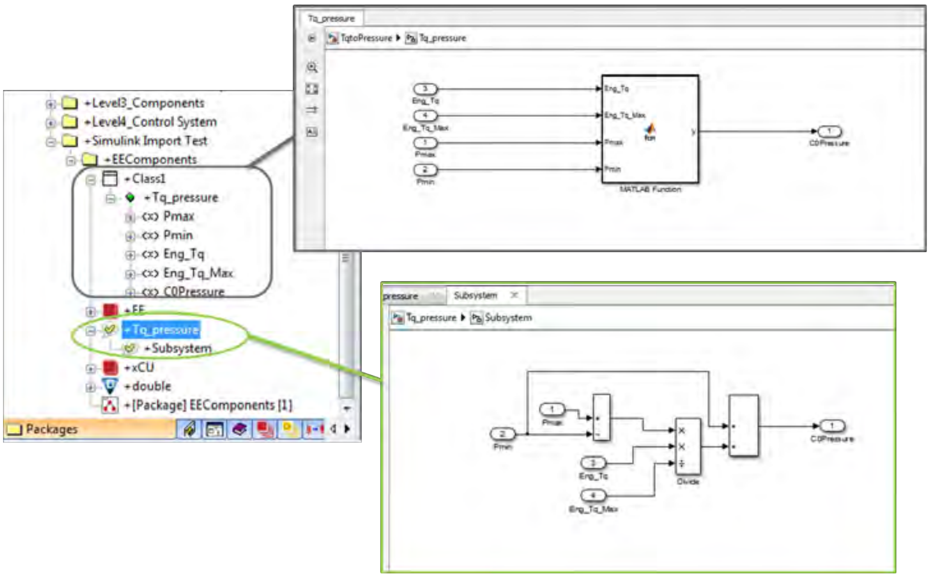

Figure A.11: Example of Class and RQ importation

CONSIDERATION:

- Simulink Profile contains a number of Visual Basic scripts that are responsible for synchronization. These scripts are customizable and expandable if necessary, so that on the particular application tailored synchronization between SysML^{TM} and Simulink is possible.
- The elaboration in Simulink models can then be used for example for model of other vehicles, without again a Simulink model must be created (reduced modeling effort and time, reusable models).

Bibliography

Atego. *SysML Tutorial v8-0-0.pdf.* http://www.atego.com/download-center/product-guide/artisan-studio-tutorials/. April 2014.

B. Heiming, H. Haupt. "Hardware-in-the-Loop Testing of Networked Electronics at Ford". In: *2005 SAE World Congress.* Detroit, MI (USA), 11-14 April 2005.

F. Sanford A. Moore, R. Steiner. *A Practical Guide to SysML: The Systems Modeling Language.* second edition.

Fowler, M. *UML Distilled: A Brief Guide to the Standard Object Modeling Language.* Addison-Wesley, 2003.

INCOSE. http://www.incose.org/. August 2014.

Italia, SMC. *Manuale di oleodinamica.* http://www.arenahome.org/dir/B%20Per%20imparare%20e%20capire/vari/musica/MANUALE_DI_OLEODINAMICA.pdf. September 2014.

M. Ehsani Y. Gao, S. E. Gay A. Emadi. *Modern Electric, Hybrid Electric, and Fuel Cell Vehicles.* Boca Raton, FL (USA): CRC Press, 2005.

N. Navet, F. Simonot-Lion. *Automotive Embedded Systems Handbook.* Boca Raton, FL (USA): CRC Press, 2009.

OMG. *OMG Systems Modeling Language (OMG SysML^{TM}) Tutorial 19 June 2008.* http://www.uml-sysml.org/documentation/sysml-tutorial-incose-2.2mo. August 2014.

— *OMG Systems Modeling Language (OMG SysML^{TM}) Version 1.3.* http://www.omg.org/spec/SysML/1.3/. August 2014.

OMG Systems Modeling Language tutorial and specification. http://www.omgsysml.org/. August 2014.

Software articles. http://www.objectmentor.com/resources/publishedArticles.html. April 2014.

T. Weck M. Maletz, P. Ebner. "Optimization of Transmission Function, Efficiency and Quality in the Development of Hybrid and Electrified Powertrains". In: *10^{th} International CTI Symposium.* Berlin (Germany), December 2011.

Volkswagen. *Electric vacuum pump for brake servo unit.* http://www.volkspage.net/technik/ssp/ssp/SSP_257.pdf. September 2014.

Acknowledgements

I would like to acknowledge my advisor Professor Giuseppe Buja and to express my special appreciation and thanks to my supervisors Dipl.-Ing. Thomas Weck and Dipl.-Ing. Martin Engelbrecht at AVL List GmbH, your advice on both research as well as on my career have been priceless. I would also like to thank all the DSS Department and other Departments of AVL List GmbH for the support during my Master Thesis, Mr. J. Fritz of Virtual Vehicle for the Model-Based SysML™ training and Atego®for providing me Artisan Studio SysML™ and then support me on my activities.

A special thanks to Cristina for her support and patience during the last six months and to my family. Words cannot express how grateful I am to my mother and father for all of the sacrifices that you've made on my behalf. Furthermore I would also like to thank Ing. Davide Da Rù, the colleague who gave a high contribute to my Master Thesis . Last but not least, I would like to thank all my friends and my university fellows for the special moments spent together , the Austrian's friends and the Italian's students and friends at AVL List GmbH.