INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN, ICED'09
24 - 27 AUGUST 2009, STANFORD UNIVERSITY, STANFORD, CA, USA

MECHATRONICS ENGINEERING ON THE EXAMPLE OF AN INNOVATIVE PRODUCTION VEHICLE

Paweł Ziemniak[1], Marek Stania[1] and Ralf Stetter[1]
(1) Hochschule Ravensburg-Weingarten, Germany

ABSTRACT
This paper presents insights concerning a design methodology for mechatronic products. In general, it follows VDI Guideline 2206 [1] and proposes the use of the well-known V-model at the macro-level and the general problem-solving cycle of systems engineering at the micro level. The V-model is customized for the mechatronic systems under development and can be used to describe the whole system (the product) or the subsystems. The paper delivers a background on mechatronics engineering in its second section and presents two mechatronic development projects in the following ones. The third section is dedicated to the product development of an Industrial Robot while the fourth section is focused on an outdoor robot called MAX 3D. Both projects originate from an innovative driving and steering solution developed in one of the previous projects. In these two sections the robot designs are presented together with occurring design problems and observations made during the mechatronic product development of these robot designs. The description of these problems and observations serves for a further development of the development methodology for mechatronic systems.

Keywords: mechatronics engineering, V-model, design methodology, robot, vehicle

1 INTRODUCTION
Nowadays mechatronic products gain more and more attention. This process is rather natural as expectations of the customers continuously rise. They want better, cheaper, and more advanced products. By more advanced products they often mean products that will have not only improved, but also additional functionalities. There are many products that are live facts of this tendency. In 1868 Ives W. McGaffey invented and patented the first vacuum cleaner. It was a purely mechanical device with a manual air pump. It was difficult to operate this device as the air had to be manually pumped during work. Years later the electric motor was used to replace the manual pump making the device not purely mechanical but electromechanical, in same sense an early mechatronic product. Few years ago simple autonomous vacuum cleaners were developed using bumping sensors and some logic. Those devices are modern mechatronic products however the capabilities of future mechatronic products are far beyond those. When will intelligent vacuum cleaners capable of traversing and cleaning whole houses be available? How to design and develop such sophisticated products?
Products that are composed of subsystems of different domains like mechanical engineering, electronical engineering, and software engineering are usually referred to as mechatronic products. Depending on particular products some fields might be developed better than others but all of them usually require specialists. Moreover, the specialists should be able to work closely together what is often difficult as they represent completely different fields of science. Usually the problem leads to different concepts of the complete solution which are specific to the main domains. This and many more reasons led to establishing a design methodology for mechatronic products. Following VDI Guideline 2206, the methodology basically consists of two procedure schemes: the general problem-solving cycle of systems engineering and the V-model [1], [2]. Stetter&Stania [3] propose to refer to the whole process as "Mechatronics Engineering".
Robotics is a field that fully exploits the term mechatronics: advanced mechanical construction, smart sensors, smart actuators, medium to large number of processing units, and sophisticated software system. Together, these items render a robot a very sophisticated mechatronic product. In The Systems Engineering Laboratory mobile robots are developed and thus an effective and efficient methodology is necessary in order to achieve the set goals. It was found that as general basis the V-model is appropriate to manage all designs in this laboratory. Besides the V-model, several other methodologies

and supporting tools are applied in order to be able to continuously improve the designs under development.
The following section explains in more details what is the core meaning of the term mechatronics and elucidates the concepts connected with mechatronics engineering. Furthermore, the V-model is described, it is discussed how it can be applied in a sensible manner, and some other supporting methods are mentioned. The theory will be then verified by practice presenting some designs carried out in the laboratory.

2 MECHATRONICS ENGINEERING

In this section the principles and tools of mechatronics engineering are explained in order to serve as a basis for the more detailed discussion on the example of the mobile robots in section 3. The term "mechatronics" is usually used to describe a combination of mechanical engineering, electronical engineering, and software engineering. This rather simple definition in fact accurately reflects what mechatronics is, but mechatronics itself is even more complicated and complex. It is rather obvious that the complexity rises together with increasing complexity of the separate domains. However, together with a higher complexity in particular domains, higher cross-domain integration is required. This integration causes an additional increase of the complexity and because of that, the mechatronics complexity is far more than just a sum of complexities of separate domains. A good example of mechatronic systems are robots. Those systems are usually characterized by elaborated mechanical construction, variety of electronic sensors, actuators, and controllers and finally, the software that runs on the controllers and takes care over all the functionality of the robot and its design into account. Developing all the three parts requires specialized engineers, but the system has to be designed first and this requires a lot of work in order to make everything fit together perfectly.
The term "systems engineering" is used to describe the process of developing sophisticated complex systems. It focuses not only on technical aspects but also on disciplines like e.g. project management. Although mechatronic products are also systems, they are of slightly different type, they are hydride systems. Because of the multidisciplinary character of mechatronics, "mechatronics engineering" term can be used to describe the process of developing mechatronic systems. Similarly to systems engineering, it concerns also aspects related to project management. Using the mechatronics engineering approach it is possible to design multifunctional, flexible, reliable, and robust products with other benefits like:

- verified functionality and fewer defects, especially with multifunctional products,
- better documentation,
- higher level of reuse between projects,
- longer life times and with simpler maintenance, and
- more precise control over cost and schedule and at the same time better fulfillment of user requirements.

Systems engineering is expected to provide similar results [4], but it is important to distinguish average single purpose systems from multipurpose and multifunctional mechatronic products which integrate sub-modules and functional peculiarities of the different domains.

2.1 V-model and general problem-solving cycle

The V-model is a graphical representation of the system development process. It was first developed in 1980s and since then it was adapted by many different industries [4]. In 1997 the model was officially adopted by German federal administration. After several modifications the V-model was suggested by VDI Guideline 2206 as a design methodology for mechatronic systems [1], [2]. It is important to note that the V-model is not a strict set of rules that have to be obeyed; instead it is an approach that not infrequently has to be tailored to specific problem. There are several researchers that try to optimize the model for specific mechatronic systems [5], [6], [7], [9], [10]. Additionally several researchers are concerned with the teaching of mechatronic design [11], [12] and the development of model, tools, and strategies for the product development of mechatronic systems [13], [14], [15].
Figure 1 presents the general structure of the V-model proposed for mechatronic systems. The shape of the model is the "V" letter and from this the name has been derived. The model shows the main flow from requirements through system design, domain specific design, and integration phases up to the final product. Vertically the model is divided into levels, which are the successive steps to be followed. During system design, first, system requirements are defined and as a result a system

specification is generated. These documents are then processed on the second level in order to obtain subsystems requirements. In the component level, a domain specific design occurs. The subsystems requirements are processed two times: firstly, in order to define components; secondly, to implement them in the specific domains (mechanics, electronics, software). In all these design steps verification occurs to check whether the design meets the requirements and if not, iterations are performed. After a successful implementation, the process of integration occurs. At first, components are integrated within the component level so that whole domain specific process is encapsulated in this level. Later, the components are integrated into subsystems and finally into a uniform system, which is the final product. In each integration phase a validation process occurs and, if some inconsistencies are found, the system design can be reprocessed from the appropriate level.

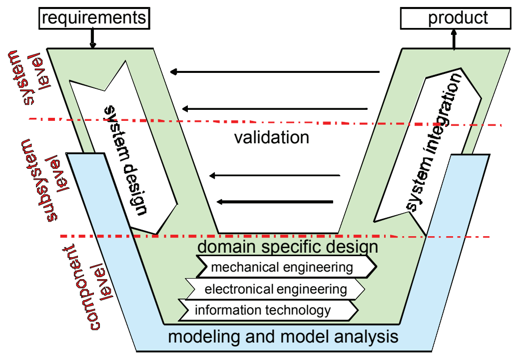

Figure 1. General structure of the mechatronic V-model [10]

Modeling and model analysis is the additional element of the V-model. It encourages using additional modeling and simulation tools in order to investigate the modeled system more precisely. The VDI Guideline 2206 recommends, besides the application of the V-model, also the use of the general problem-solving cycle of systems engineering in its methodology for mechatronic systems.

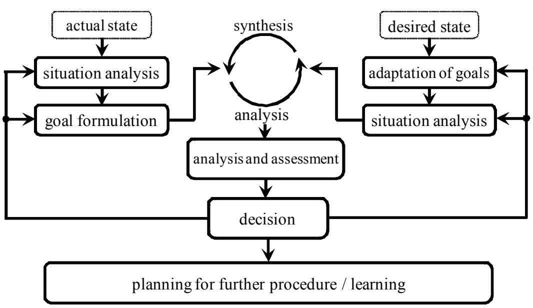

Figure 2. General problem-solving cycle

The V-model is present on the macro-level and concerns the whole system or complete subsystems. The general problem-solving cycle is present on the micro-level and is used in individual steps inside the macro-level. The term micro-level is just an agreed term, the tasks within can last several hours as well as several months. The problem-solving cycle methodology was adapted from systems engineering [16] and is a guideline for the problem solving process also in mechatronic systems. In the general problem solving cycle, several steps are cyclically repeated in order to find an optimal solution for the problem (see Figure 2). During those steps the state of the art is analysed and goals are formulated (left side). Additionally the current state is checked for compatibility with the desired state (right side). Then, during synthesis, analysis, and assessment steps, the improvements are incorporated and the decision is made whether the state of the art is the desired state or not, if not the whole cycle is repeated.

2.2 Further improvements focused on the V-model

In more sophisticated mechatronic systems the V-model might not necessarily represent the whole mechatronic product development process; instead, it can represent particular product lifecycle stages. In such situations a set of sequential V-models can be used. With increasing product maturity the additional (outer) V-models are carried out and evaluated. The results of the consecutive integration steps can be e.g. product requirements, prototypes, and finally the product.

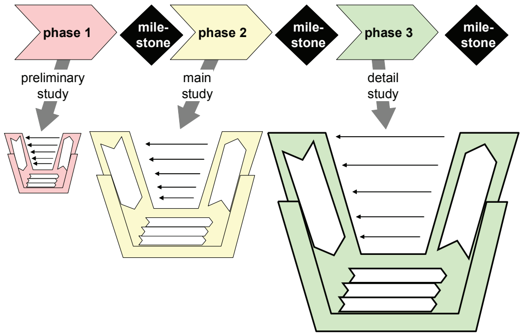

Figure 3. Possible relations between project plan and process maps (V-models)

The V-model is often misinterpreted as a tool that provides a complete methodology and all the project management necessary to develop requirements into the market ready product. In fact it does not provide even such mechanisms like schedules and milestones. It is possible to establish milestones, but it is not the main point of the V-model and usually pure V-model related milestones might be to general or inadequate. Furthermore, many authors [17], [18], [19] point out that the product development is not a sequential process. The development is usually characterized by iterations and jumps between certain stages. It is essential to distinguish a tool that is mainly designed to provide a design methodology from a tool designed for project management. The V-model shows the logical way from requirements to a market ready product, but, as an analogy, it is only the map that shows the checkpoints that had to be followed but not their sequence and timing. Figure 3 presents how the V-model can be related to the schedule. In fact, there are several V-models and completion of each is an achievement of the milestone. This is not the only, but one of the most usual ways, as mechatronics products are too sophisticated to put everything into single model and define milestones only within it.

3 INDUSTRIAL ROBOT

Figure 4 shows the chassis of the production vehicle currently under development called “Industrial Robot”.

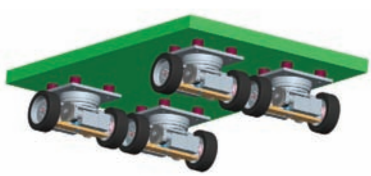

Figure 4. The chassis of the Industrial Robot

The idea of an Industrial Robot, that is appropriate for heavy transport and disposes of high maneuverability, was raised during the evaluation of the innovative steering solution developed at The Systems Engineering Laboratory. This steering principle was proven in the first mobile robot called

Max 2D. The mechanical design of Max 2D consists of four identical driving units. The design of the units allows the use of the same, single motor for propulsion and for steering. It means that there are only four motors in the robot and all of them are used for propulsion as well as for steering purposes. The central positive characteristic of the robot is that it can follow any 2D direction. The University applied for a patent for the solution and the concept was successfully implemented in the mobile robot Max 2D.
Current work focuses on a new design, which is called Industrial Robot, and on using a similar steering solution. Like in the previous design, also in the new one it is possible to follow any 2D direction. Figure 5 presents the vehicle model as seen from bottom. Four driving modules are present; each is equipped with two wheels. These modules align properly to the direction of the movement.

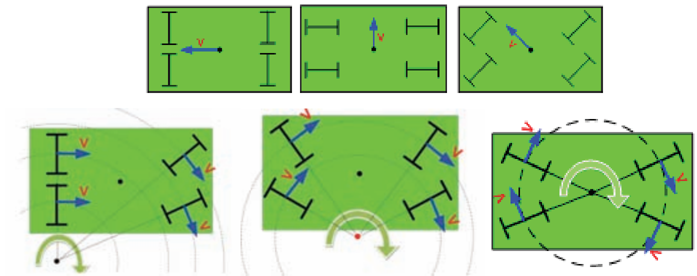

Figure 5. Possible moving directions

In Figure 5 the first three depictions presents the vehicle while following any 2D line, the next depictions presents a car-like turning (Ackermann steering), no slip turning, and finally central point rotation. The new mechanical design also consists of identical driving modules. Such an approach reduces the development, manufacturing, and maintenance cost (and time) and enhances robustness. Furthermore, for special purposes, the number of modules can be increased raising the total power of the vehicle and even allowing sophisticated platform shapes as opposed to basic rectangular shape.

3.1 First concept of the system

Figure 6 presents the main V-model template that was used for development of the robot. The template is compatible with general structure of the V-model. The domain specification starts at the subsystem level.

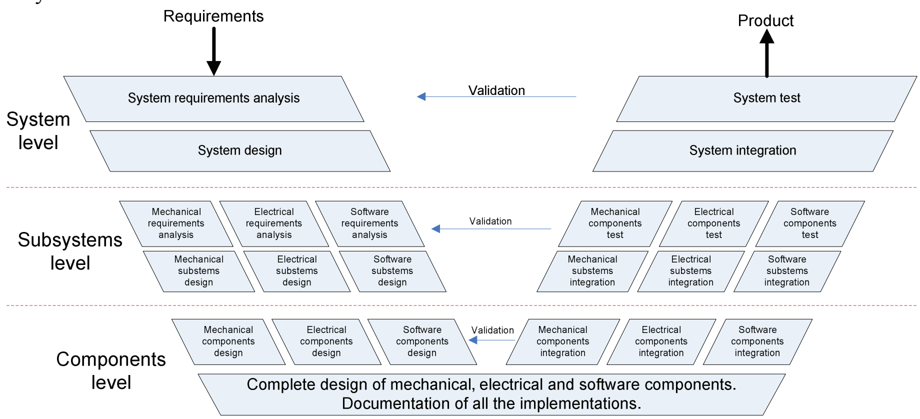

Figure 6. Used template for the V-model

The template was first used for the analysis of the requirements given to the system (the robot), which first resulted only in a rough concept. Main requirements were related to industrial needs for autonomous transportation systems. It was essential to design a unit that was fast and could operate in limited spaces e.g. narrow passages between machines. The project had one specific requirement - it had to develop the a priori idea of four wheel steering robot characterized by no dedicated steering motors (only propulsion motors). Some simple requirements of the system were put into the requirements lists. They were usually countable requirements like e.g. maximum velocity, acceleration etc. Both the demanded value and the tolerance were provided in the list. These requirements were also assigned to the related domains assuming that one requirement can be of significance to one or more domains. Most of the requirements, however, were put into UML use case diagrams. UML (Unified Modeling Language) is a modeling tool well known from software engineering, but successfully adopted also into other domains. The advantage of the use case diagrams is mainly the simplicity of presentation; there is a system or system component and actors that make use of its functionality. Most of the requirements could be shown in an abstract manner that is understandable even to non-professionals. According to the V-model, the system level should be characterized by high level of abstraction in order to allow cross-domain solutions to arise. Using these diagrams helped mechanic, electronic and software engineers to fully understand each other and to analyze together what exactly is to be done. It is extremely important to have a global view at this point, this allows developing real mechatronic products; otherwise more domain specific solutions could arise where some domains could be better developed in cost of others or some domains might be limited by others. As an example, a well developed but computationally expensive control system could not work optimally on a hardware with limited parameters. In order to succeed with the development, at least one whole domain has to be redeveloped to comply with the others. There are many more examples of links between the domains and keeping the solution abstract and unrestricted by domain-specific thinking is a fundament of the early stages of the V-model.

After several evaluations of the rough concept of the system, a final process map was prepared. It depicted what had to be achieved and how to achieve it. The base of the map was the V-model of the whole system with additional V-models embedded as subsystems. This way more advanced subsystems, the one that were strictly mechatronic i.e. with strong links between all the three domains, were distinguished.

The problem with the application of the V-model was that it is definitely not a sufficient tool for the whole project management. It has to be supported by other tools, e.g. the well known Gantt chart to schedule the development process. During the development of the Industrial Robot several V-models for different stages of the development were prepared. Some points on the Gantt chart were directly related to the completion of some of the V-models (similarly to Figure 3) and some were even related to the completion of stages within the V-models (e.g. the design or integration of a subsystem or component).

Besides time and resource management, it is also essential to provide documentation in a uniform format that can be easily accessed and analyzed by specialists of all domains. CAD models were used to design all the mechanical parts and to put electronic devices inside, also the wiring of the system was proposed on appropriate diagram. The software was modeled entirely using UML: use case diagrams, sequence diagrams, and class diagrams. The class diagrams played also a major role in creating the V-models; the classes represented subsystems and components. The hierarchy of the whole system was shown by inheritance; properties were reflected as fields, and abilities as methods. The UML, although by many people criticized by being large and complex, is a tool which basics can be easily understood and basic diagrams are often sufficient and well understandable. Those additional modeling tools played a significant role in the development process: they were easily understood by engineers of all the domains, allowed communication on an abstract level, and left the documentation of all the steps taken in the project.

3.2 Robot subsystems, the driving module

The second level of the V-model is the subsystem level where subsystems are specified and at first theirs requirements are considered. This results in designs of all the subsystems. According to the template (Figure 6) the domain specification starts in this level. In simple systems the subsystems can be directly assigned to specific domains, in our project however, some subsystems are highly dispersed into the three main domains and it was reasonable to model them explicitly in new instances of the V-models. There are additional advantages of such a solution. The driving module, for example, is one of the main components of the robot and has innovative steering principle which is expected to evolve over time and might lead to future innovations. Putting the driving module into a separate

model gives better insight into this part of the system and increases the ability to reuse it in different projects with minor or major modifications or improvements.
The design of a single driving module is presented on Figure 7. The CAD model consists of two wheels and two motors and thus it is similar to a differential drive. Thanks to the bearing in the upper part, the lower part of the module is able to rotate freely while the top of the module is fixed to the platform. In order to provide unbounded rotation, a slip-ring is used to connect the in-module electronics with the rest of the vehicle.

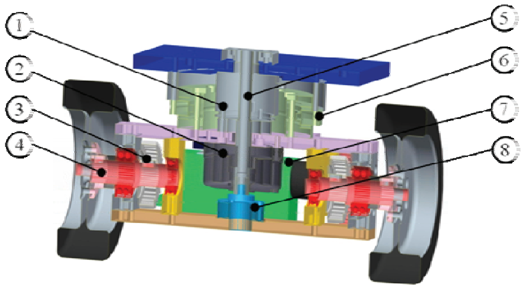

Figure 7. Main mechanical components of the driving module: 1 - encoder, 2 - brake, 3 - gearbox, 4 - horizontal axis, 5 - vertical axis, 6 - bearing, 7 - motor and controller with gear head, 8 - slip-ring

Besides motors and motor controllers (7), the module consists also of an electromagnetic brake (2) and an encoder (1). Both of them work with regard to a vertical axis (5), the former is used to block the rotation while the latter measures the angle of rotation. This angle represents also the direction of movement of the vehicle. The module acts like a differential drive and in normal operation the angle is maintained by torque differences on the wheels. In failsafe mode however, the brake can be used to help fixing and holding the angle e.g. when one or both motors fail.
The solution would be difficult to arise within a single domain as, in fact, it is a composition of all the domains. There is no motor responsible only for steering the wheels (like in conventional vehicles), instead, thanks to special mechanical construction and appropriate control software, the propulsion motor can be used for that purpose. Simply saying, the motor was dispersed into special mechanical design, additional software, and additional electronic device - a brake.
All the electronical components are interconnected by means of a CAN bus making it easy not only to configure and replace them, but also to supervise and log the communication process for security and reliability purposes. The devices were carefully selected in order to comply with the requirements inter alia: available space (encoder put inside the bearing), limited throughput of the slip-ring (the motor units are in the rotational part of the module and integrate motor, encoder, and motor controller in a single mechatronic device with a CANopen interface). The process of selecting the devices played a major role and required both an abstract view at the beginning to properly state what is required and to design it, the domain specific views for selecting particular devices and solutions, and finally the integration phases for analyzing compatibility of the choices. The process was additionally encapsulated into the general problem solving cycle in order to sequentially improve the outcome what resulted in even higher cross-domain integration.
The driving module by itself is already an elaborate mechatronic product. Starting from the requirements of the driving module a cross-domain solution was prepared and was later dispersed into subsystems assigned to specific domains, and further into components. Next, according to the V-model, the complete design of all the components was made followed finally by integration on the consecutive levels. On each integration level a validation was performed in order to check whether or not the requirements of the components, subsystem and finally the system (module) are met. The validation is an important step and allows early detection of possible failures. If a component or subsystem does not meet the requirements, it is obvious that the same will happen with the system and the V-model has to be reprocessed from appropriate point. Several more subsystems are present like the mechanical design of the robot frame, power supply, battery charging, robot control, sensory etc. The last subsystem can also be considered using a separate V-model. It consists of different sensors (mechanics and electronics) and system that recognizes environment from the readings (software). The

advantage of using a separate model here is reusability. The sensory solution and its documentation can be easily copied and pasted into new projects with or even without additional evaluation of the V-model, depending on conformity of the requirements in both projects.

3.3 Prototypes

As written in the previous section, the project is described by several V-models. The V-models are not only embedded into the main V-model, but are also placed in time similarly to the template presented on Figure 3. The first models helped revising the innovative steering concept and designing possible paths of improvements. Finally, a model was prepared to design and build the simplified prototype. It was more convenient to test the partial design before building the complete robot with four identical driving modules (and duplicating the possible problems within single unit). Figure 8 presents both the first revision of the driving module model and the testing platform (first prototype) with that module.

Figure 8. The first revision of the module and the testing platform

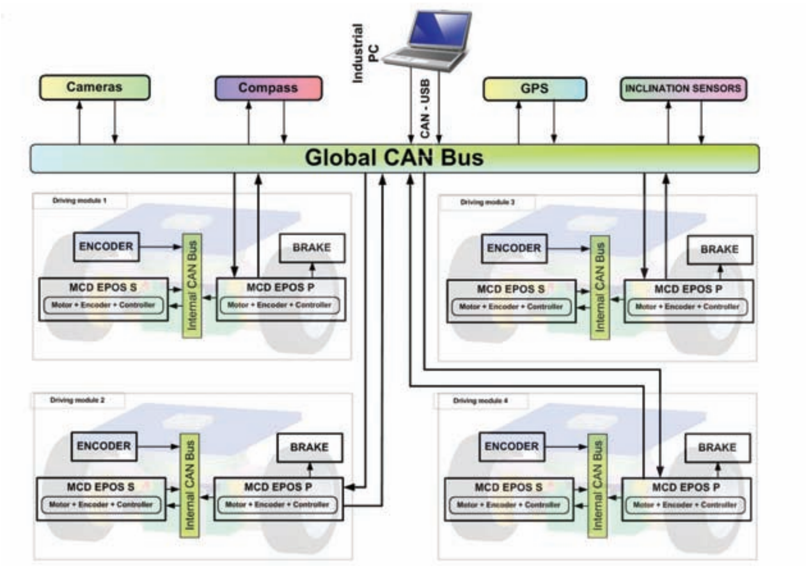

Figure 9. The control system architecture

The first performed tests revealed several problems, which mainly concerned the driving module. Thanks to the early verification of a not yet complete solution, but of just one subsystem, it was possible to quickly solve the problems and redesign the rest of the driving modules before manufacturing them. Thanks to this procedure a lot of time and money was saved. Only the V-model corresponding to the driving module needed to be evaluated. Thanks to the tests of just one single module, it was also possible to find future bottlenecks of the control system architecture. The tests revealed that the two motors and the encoder consume a lot of network throughput in order to provide smooth operation. Extrapolating it into four modules, it was found out that in the final product, with additional devices interconnected by means of the CAN bus, major communication problems might occur. In order to omit this issue several CAN buses were incorporated into the solution as presented on Figure 9. The idea was to have separate in-module buses that interconnect the motors and the encoder and that are also connected to global CAN bus. This way some messages are kept within particular modules and the bus load is distributed.

3.4 Summary and future research

The continuous monitoring, reflection, and evaluation of the projects in The Systems Engineering Laboratory, even the finished ones, allows both an improvement of the old ones and finding new ideas. Thanks to that approach, improvements to the innovative driving and steering solution of the mobile robot MAX 2D were possible and resulted in the Industrial Robot project. The driving module was one of the main subsystems and as such it was modeled on a separate V-model embedded into main V-model of the whole system. Such an approach gave the possibility to work more intensively on it and to consecutively develop improved versions. The strategy of developing and immediately testing was applied in order to foster an early determination of product properties (compare [20], [21]). The early testing allowed not only finding current, but also future problems. The special platform, the first prototype, was build in order to provide an appropriate test bed for the first version of driving unit. The cost was much smaller than the cost of redesigning all the modules and the prototype of final chassis. In the sense of systems engineering, the realization of the first module can be understood as a preliminary study (compare [7]).

The first prototype chassis with a single module behaves slightly different than with two modules and the chassis was sized to accommodate up to two modules. Recently, the second, improved version of the module was build and will be tested until the redesign and reconstruction of the first version. After this, the platform will be slightly redesigned in order to accommodate two modules, this time mostly the global V-model will need to be reevaluated while the sensory system might be left untouched and driving unit might require only minor modifications. After the successful testing and possible redesigns, the next modules and eventually the final chassis will be designed and produced according to the project schedule.

4 THE DESIGN OF MAX 3D ROBOT

The second design currently under development in The Systems Engineering Laboratory is an outdoor robot called MAX 3D. The mobile robot is developed to be able to drive on rough terrains and over relatively big obstacles. The mechanical design again consists of four identical driving modules. Such an approach proves to have many advantages: the single unit can be analyzed more deeply, is easier to develop and manufacture, and finally maintenance is easier (e.g. replacing the damaged component).

The requirements for the robot were similar to those of the Industrial Robot and a similar template of the V-model was used. The additional requirement for the robot was to drive on rough terrain or over obstacles while keeping the demanded platform inclination. Most of the subsystems were transferred from the Industrial Robot only with some minor customizations. The driving unit and whole control system, however, had to be fully redesigned in order to advance into 3D space. Moreover, during the evaluation of the early V-model of whole system (preliminary studies) a possibility of compensating external forces by inclination of the robot chassis was observed. The hardware can be enriched by accelerometer and software can be added to detect and compensate centrifugal forces or acceleration/deceleration by an appropriate inclination of the robot chassis. The software system will require a dynamic model in order to be aware of all the important forces acting in the system.

4.1 The architecture of the control system

The design, since distinguished by an additional degree of freedom (3D space), requires a more advanced control system. The fact that the steering, velocity, and level control is supported only by one motor, and brakes, and encoders, directly increases the complexity of the control system.

Building such a system is a difficult task and also requires an appropriate approach for development. Here, as it is an example of the domain specific design, software engineering comes with a helping hand. One of the basic rules of the software engineering is the modular structure whether it is structural or object oriented programming. According to this rule the software system was divided into the two main levels (Figure 10): the low level control responsible for steering, velocity and level control, and the high level control responsible for navigation, path planning, and path optimization. Going further, the levels are divided into layers which are the modules strictly responsible for the given functionality; Figure 10 presents also the layers of the lower control level.

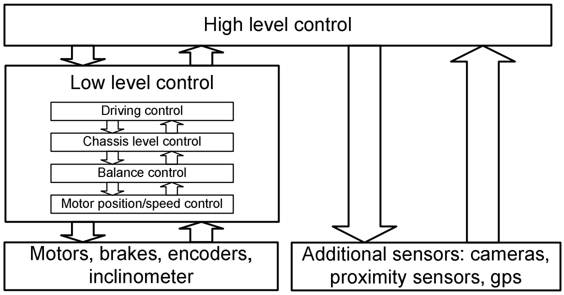

Figure 10. Architecture of the control system (source: [8])

It is important to note that the higher level works in abstraction from the motors, the brakes, etc. Such abstraction allows reusing the higher level in different projects providing that the feasible maneuvers and interface of the low level control are compatible, otherwise some modifications might be necessary.

4.2 The design of the mobile robot MAX 3D

The mobile robot MAX 3D was developed in two versions. Figure 11 a) presents the first one and Figure 11 b) the second one. The latter was a result of solving a problem with turning when the robot was heavily inclined. The problem arises because, in order to steer the robot, the wheel has to travel longer distances in such situations.

Figure 11. a) Chassis, b) alternative chassis, c) driving module of the mobile robot MAX 3D

The general problem solving cycle had to be carried out here rather on the macro level as a major modification not only of the module but of the whole system concept was necessary. After reverting and reevaluating one of the earliest V-models of the system a decision was made to bend the wheel-ended legs and bring the motor down loosing however the ability to rise the leg in the air in front of the obstacle. During evaluation it was shown that the system requirements were badly analyzed and such an ability of the robot is not crucial. It is more important to steer the robot smoothly and only to run over the obstacles. Raising the leg is optional and rather unstable operation since the robot has to stay only on three wheel-ended legs.

Figure 11 c) presents a detailed design of the alternative version of the driving module. Also here, the devices had to be chosen carefully for the provided rigorous restrictions - weight, cabling capabilities, power, dimensions, etc. The devices are of the same type as in the Industrial Robot, here an additional brake and encoder are present due to the additional degree of freedom. In the vertical axis (1) a brake and an encoder are provided. On the horizontal axis (5) an encoder (2), a brake (3), and springs (4) are present. The motor (with embedded controller) (7) is mounted directly at the lower part of bended leg (6) and the propulsion is transferred to the wheel (8) by means of angular gear-box (9). The platform level can be changed by virtually stretching the vehicle by means of an appropriate velocity control. The springs help to control the level and provide some level limitations; they are also helpful in decreasing trembling between suspension and the platform of the robot. The brakes are used to help controlling and then holding the level or inclination.

4.4 Summary and future research

Thanks to the well suited and documented project management of the Industrial Robot project, during the development of MAX 3D it was possible to easily make use of the same templates and tools. Moreover, a number of subsystems and components together with their documentations were easily transferred and adopted to the new project. Even some elaborate subsystems needed only small adaptations e.g. the sensory subsystem: its separate V-model was started by adding requirements regarding reading the robot inclination, what resulted in adding an inclination sensor, some wires and appropriate fragments of software. The reusability of subsystems and components dramatically reduced time and resources necessary for the project. Without modular and transparent design, the adaptation process would prolong, and sometimes might come to an extent that is no longer profitable.
This design is also an example of wrongly interpreted requirements what unfortunately happens in many projects. The requirements should be processed carefully on each step of the design during the design phase and the parts of the system should be validated and compared to the requirements during the integration phase.

5 SUMMARY

The development of mechatronic products is a highly elaborate process. Mechatronic products are complex systems that widely expand to mechanical engineering, electronical engineering, and software engineering. The complexity of such systems is far more than just a sum of complexities of the main domains. It is essential to keep in mind that the domains have to be smoothly composed as otherwise the product might not fulfill the set requirements and might not meet user expectations. It is also important that the user expectations continuously rise, they want better, more advanced products and at the same time more inexpensive. In order to fulfill the requirements and reduce development costs, a good and verified methodology is essential. Following VDI Guideline 2206, the V-model is the preferred process map, but additional tools are essential in order to manage the project and develop the requirements into market ready product in an effective and efficient manner.
In the second part of the article the two designs were presented. They both originate from exploration of a mobile robot called MAX 2D which was developed in The Systems Engineering Laboratory. In these projects the innovative and patented driving and steering solution was redesigned in two ways. In the first project, called Industrial Robot, the power and reliability was increased, while in the second one, a mobile robot called Max 3D, an additional degree of freedom was gained allowing chassis level regulation. Both projects share several subsystems and use the same methods and templates for project management. A lot of time and resources was saved this way and it would not be possible without transparent design, hierarchical system structures, and clear documentation at each step. The V-model played a major role but during the design and realization of the mobile robots it was also learnt that:

- The use of a V-model as a logical structure has to be accompanied by additional tools for project schedule management.
- A single V-model, even evaluated repeatedly, is often not enough for complex mechatronic systems; several V-models might be necessary in order to develop from the requirements into market ready product allowing prototypes to be easily derived all the way. The prototyping is not only important for elaborate systems, but also for developing innovative, not yet well known solutions.
- Hierarchical structuring of the V-models allows better insight into the system and through better decomposition provides also better reusability of the subsystems and the components.
- The general problem solving cycle, although proposed and mostly used at the micro-level, can be also beneficial at the semi-macro or even the macro-level. By continuous monitoring and improvement it can help finding optimal solutions even of elaborate problem, or help solving a problem that lies in the grounds of the project and requires major changes in the approach (like in section 4.2).

- The early testing of even partial solutions often helps determining final product properties and prevents possible future failures.

Future research focuses on the further development of the Industrial Robot and MAX 3D. The MAX 2D project, although it might be assumed finished, is in continuous evaluation as might result in some improvements or clues that might help improving others or starting new projects, besides it serves as a test platform for various sensors and higher level software such as environment recognition.

REFERENCES

[1] VDI 2206: "Design methodology for mechatronical systems", Beuth, Berlin, 2004.

[2] Gausemeier, J., Möhringer, S.: „New Guideline VDI 2206 – A flexible procedure model for the design of mechatronic systems", Norell, M. (Ed.): Proceedings of the 14th International Conference on Engineering Design (ICED)", Stockholm, 2003.

[3] Stania, M., Stetter, R.: "Mechatronics Engineering on the Example of a Multipurpose Mobil Robot". In: Solid State Phenomena Vols. 147-149 (2009) pp 61-66.

[4] U.S. Department of Transportation: "Systems engineering for intelligent transportation systems: An Introduction for Transportation Professionals". U.S. DoT: January 2007.

[5] Bathelt J., Jönsson A., Bacs C., Dierssen, A, Meier, M.: "Applying the New VDI Design Guideline 2206 on Mechatronic Systems Controlled by a PLC". In: Proceedings of ICED05 International Conference on Engineering Design, Melbourne, Australia, 2005.

[6] Jansen, S.; Welp, E.: "Model-Based Design of Actuation Concepts: a Support for Domain Allocation in Mechatronics". In: Proceedings of ICED05 International Conference on Engineering Design, Melbourne, Australia, 2005.

[7] Gausemeier, J.; Giese, H.; Schäfer, W.; Axenath, B.; Frank, U.; Henkler, S.; Pook, S.; Tichy, M.: "Towards the Design of Self-Optimizing Mechatronic Systems: Consistency between Domain-Spanning and Domain-Specific Models". In: 16th International Conference on Engineering Design (ICED'07), August 28-31, 2007, Paris, France, 2007.

[8] Ziemniak P.; Uciński D.; Paczyński A.: "Robust Control of an All-Terrain Mobile Robot". In "Solid State Phenomena" Vols. 147-149 pp 43-48, Switzerland, 2009

[9] Bernardi, M.; Bley, H.; Schmitt, B.: "Integrating a Mechatronics-oriented Development Process into a Development Department". Proceedings of the 37th CIRP International Seminar on Manufacturing Systems, pp. 265-270, Budapest, Hungary, 2004.

[10] Voos, H.; Stetter, R.: "Design and Control of a Mobile Exploration Robot". In: „Proceedings of Mechatronics 2006. 4th IFAC-Symposium on Mechatronic Systems". Heidelberg, 2006.

[11] Welp, E., Labenda, P., Jansen, S., "Teaching theory and practice in mechatronics engineering", Proceeding of the International Design Conference-Design 2006, Dubrovnik, Croatia.

[12] Fan, Z., Detlef, M., Andreasen, M., Hein, L., "Teaching system integration of mechatronic system", Proceeding of the International Design Conference-Design 2006, Dubrovnik, Croatia.

[13] Czichos, H.: „Mechatronik", Wiesbaden: Vieweg, 2006.

[14] Alciatore, D.G.; Histand, M.B.: "Introduction to Mechatronics and Measurement Systems", New York: McGraw-Hill, 2003.

[15] Roddeck, W.: „Einführung in die Mechatronik". Stuttgart: Teubner, 2003.

[16] Daenzer, W.F., Huber, F.: „Systems Engineering – Methodik und Praxis", 1997.

[17] Ehrlenspiel, K.: Integrierte Produktentwicklung. 2nd Edition. München: Hanser, 2006.

[18] Lindemann, U.: Methodische Entwicklung technischer Produkte. Methoden flexibel und situationsgerecht anwenden, Berlin: Springer, 2007.

[19] Pahl, G., Beitz, W., Engineering Design: A Systematic Approach, Springer-Verlag, Berlin, Heidelberg, New York, Tokyo, 2006 (2nd edition).

[20] Bernard, R.; Stetter, R.: Early Determination of Product Properties. In: HUBKA, V. (Hrsg.); et al.: Proceedings of ICED 97, Tampere. Zürich: Edition Heurista, 1997, S. 2/675 - 2/680.

[21] Lindemann, U.; Stetter, R.: Industrial Application of the Method „Early Determination of Product Properties". In: LIPKIN, H.; MISTREE, F. (Eds.) Proceedings of the ASME 1998 Design Engineering Technical Conferences and Computers in Engineering Conference. Atlanta: ASME International, 1998, CD-ROM.

Contact: Prof. Dr.-Ing. Ralf Stetter, Hochschule Ravensburg-Weingarten, stetter@hs-weingarten.de