HIGH LEVEL SOFTWARE SUPPORT OF AIRCRAFT- AND DESIGN SKILL DEVELOPMENT

György Bicsák
PhD Student
BME Department of Aircraft and Ships
H-1111, Budapest
Stoczek st 6
Building J. 4th floor 418.
gybicsak@rht.bme.hu

István Jankovics
PhD Student
BME Department of Aircraft and Ships
H-1111, Budapest
Stoczek st 6
Building J. 4th floor 418.
ijankovics@rht.bme.hu

Abstract. The students specialized on vehicle engineering and aeronautical branch must study several special courses about aircraft design. These subjects and generally the curricula include three major parts: aircraft development philosophies, aircraft design and aircraft systems- and parts design. Of course, students preliminary have passed the subjects like mechanics, stress analysis, aerodynamics, flight mechanics, etc., because without these knowledge the basic fundamentals wouldn't be understood.
The Department has got several important and high level software, for example MATLAB with many toolkits; ANSYS Workbench, which can be applied to CFD and FEM analysis even in multi physical simulations; LabVIEW to monitor, automatize and digitalize experimental measurements; and specialized aircraft design software like ADSMaker applying to small aircraft design or AAA - Advanced Aircraft Analysis produced by DAR Corporation of professor Roskam. The Department develops its own software for solving several particular tasks, like investigation of using MEMS technology for flow control, optimization of the airfoils using inverse design methods, or developing flight control laws that immediately can be tested in department flight simulator. The students have to use engineering software in their studies for different problems. About 60 % of students use high level engineering software for their thesis. The PhD students not just use, but develop software for different applications.
The paper shortly describe the software using technologies in education processes and show some results of applying high level software to solve aircraft development and design and research projects.

Keywords: high level engineering software, teaching method, practical applications

1. Introduction

Nowadays, the influence of technology to the education is higher as it has ever been. The new generation achieves higher computing knowledge during their childhood, and has got higher expectations that university will introduce them to appropriate technologies for their successful transformation into industry. [7.]

Recently, the incredibly growing computing capacity allows using computers in many fields of engineering. From other viewpoint: there are less and less fields in science, where the computers are not used. Today, a just graduated engineer should be able to use several high level engineering software, if want to get a job. Of course, these software are just tools for design and technical preparation of production; it is indispensable to handle them with professional skills knowing the relevant physical fundamentals.

Experimental activity develops a strong connection between theory and reality by the observation of concepts and principles at work. This is instrumental for building a mental model through the construction of meaning during the learning by doing process, and for achieving deeper knowledge together with the ability to solve practical problems. Experimental activity is fundamental in many teaching fields and especially in the engineering area. It gives a priceless contribution to the Bologna student centred concept, by fostering student autonomy in the process of acquisition of new skills, competences and knowledge. [4.]

2. Teaching Methodologies

The teaching methodologies applied within engineering software education depend on a wide range of factors: tradition of teaching, the motivation and the preliminary achieved knowledge (in mechanics, fluid flow, mathematical principles, chemistry etc.), the relationship between theoretical and practical aspects of education, the teacher's specific teaching style etc. Even though the university community adopted the object-orientation as an appropriate paradigm with strong expectations and opportunities, it appears that it is not a particularly easy task and needs a lot of effort to adjust the teaching style and methodology appropriately. [11.]

The traditional teaching method of engineering software would be the typical face-to-face lectures completed by assigning students designing problems and projects to solve. The problem with this method is that the curriculum doesn't allow as much course hours, as it would be necessary. At the same time, the wide range of engineering software using skills would be necessary for the graduated engineers, at least in basic level. Another factor, that influences the teaching method significantly, that the level of engineering software development and the supporting hardware are continuously changing: new products are launched, production management system, equipment, technological processes etc. change persistently. Therefore the system of universities can't provide an individual with the necessary knowledge during the whole professional life. [5.] The purpose is to teach the engineering students a high-level, wide-range useable, problem solving knowledge, which can be used during the professional activities.

The Department of Aircraft and Ships tries to keep up with these requirements, and integrates the software teaching into the professional courses. In that way, it becomes possible to point at the relationship between the theory and practice.

For students with more talent and ambitions, the Department provides opportunity to be a part of industrial development or research projects, where besides the knowledge they can learn the importance of keeping deadlines and in some cases can see the result of their work. For the students, who works on their diploma thesis, generally if it possible a part of an industrial or research project is offered. The efficiency of this method is very high, and at the same time getting a job after graduation becomes much easier, because there is a relationship between the industry and students, before their graduation.

In the following chapter some software are introduced, which used and taught by the Department, and some examples from their applications.

3. Software and applications used in education of aerodynamics, flight mechanics

The Aircraft Design Software (ADS) is a useable tool, developed by Optimal Aircraft Design Ltd. [21.]The software is available in 5 different packages depending on the purpose of usage. OAD offers these different packages with different functionality and modules. The Professional and education version provide the same functions and modules. These versions have got optimization analysis modules, possibility to export 3D geometry into CAD files, etc. The Department has 20 network licenses of ADS for education purposes. These licenses can be used by students to realize their own concepts and ideas for designing a small aircraft. In a simple way, they can define the geometry, weight and other required parameters of the aircraft, and the software calculates the aerodynamic and flight characteristics of the designed aircraft within a few minutes. Although, they have learnt and are learning different calculation methods while they study Aerodynamics and Flight Mechanics courses, it still gives an excellent opportunity to realize a lot of concepts virtually within a relatively short time, analyse the results, and test it in a unique way in a licensed copy of X-plane9 flight simulator. It interactively helps them in the early stages of their study, when they do not have got deep knowledge in the earlier mentioned subjects yet, to understand better what different changes on aircraft mean and cause to the flight performance, handling qualities, etc.

The Department has a license of Advanced Aircraft Analysis (AAA) [22.] software, produced by DAR Corporation, founded by Dr. Jan Roskam. It is well a known software around the world. Using this software is more complex than the previous one, but it lets the students to examine their concepts in more detailed way. Because of the complexity, it requires more theoretical knowledge. The results of their work with this software can be used directly or can be implemented in the flight model of any simulator for further investigations.

When the former Corvus Aircraft Ltd. in collaboration with the Department of Aircraft and Ships designed the Corvus Racer 540 aerobatic aircraft, for perform aerodynamic design tasks the development of software was needed, instead of buying another expensive software. [8.] The idea of self-made code lived up to expectations, because this method provided the best solution from every point of view. The whole design process could be controlled by neglecting those factors, which don't influence the results significantly, or do not have at all, and emphasize those factors, which are really peculiar for aerobatic and race aircraft (simplified wing geometry, small or zero sweep angles, no twist, no dihedral, profiles with high relative thickness, no flaps, low aspect ratio, high power/mass ratio, fixed landing gear, short fuselage, lower stability, etc.). The code was developed in MS EXCEL with VBA (Visual Basic for Application), because EXCEL gives useful "GUI" by nature, and VBA provides the opportunity to write custom interpolation routines, linear equation solver, etc., nevertheless MS Office or Open Office can be found on almost every computer, thus buying any other software was not necessary to run this algorithm. On the other hand, the code can be freely modified on user's discretion. Students can use and modify it, when they are doing their own projects. In that way they can learn how general software, like EXCEL, with some self-developed code can be used in engineering practice in easy tasks instead of purchase a commercial software.

MATLAB and Simulink are produced by Mathworks inc. and are perhaps the most widely known software package in the world today in many field of sciences. MATLAB is a high-level language and interactive environment for any numerical computation, visualization, and programming. With MATLAB, the user can analyse data, develop algorithms and create models, applications. The language of MATLAB, available tools and built-in math functions enable to the user to examine multiple approaches and reach a solution faster, than with any traditional programming languages, such as Pascal, Delphi, C/C++, Java, etc. [23.]

Simulink is a block diagram environment for multi-domain simulation and Model-Based Design. It supports system-level design, simulation, and continuous test and verification of embedded systems. Simulink provides a graphical editor, customizable block libraries, and solvers for modelling and simulating dynamic systems. It is integrated with MATLAB, enabling the user to incorporate MATLAB algorithms into models and export simulation results to MATLAB for further analysis. [23.]

Using MATLAB requires basic knowledge of programing, but not as much as any other programing languages. It is easy to use, and students can focus on their tasks instead of solving data management, variable declaration and incompatibility problems. It is an efficient tool for higher level tasks. The major applications with MATLAB in the Department are supporting and creating flight simulations. In collaboration with former Corvus Aircraft Ltd. a flight simulator was developed in Simulink with S-functions to simulate the motion of designed aerobatic aircraft in real time, as it is flying along the race course. Many possibilities exist to represents the flight in 3D environment. Simulink can be connected with aerospace toolbox to Flightgear, or with Aerosim toolbox to any version of Microsoft Flight Simulator. In most cases the MS Flight simulator 2004 is used in the Department's simulator laboratory. The required aerodynamic, performance, etc. parameters of the

MATLAB simulation is provided by the self-developed EXCEL application, which was earlier described.

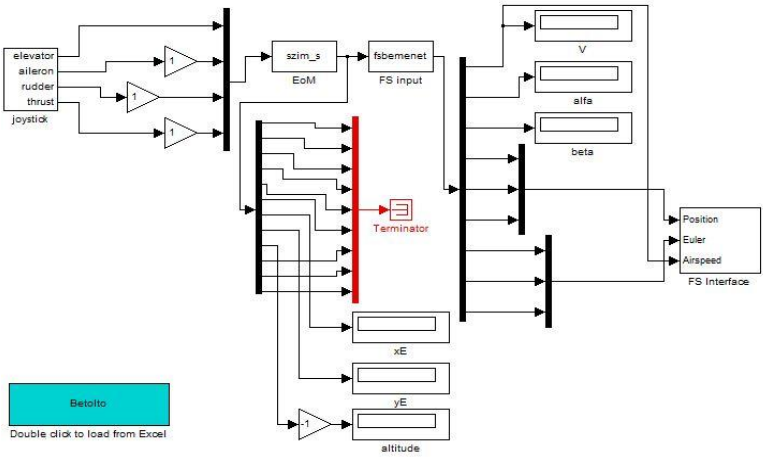

Figure 1: Simulink model using S-function blocks

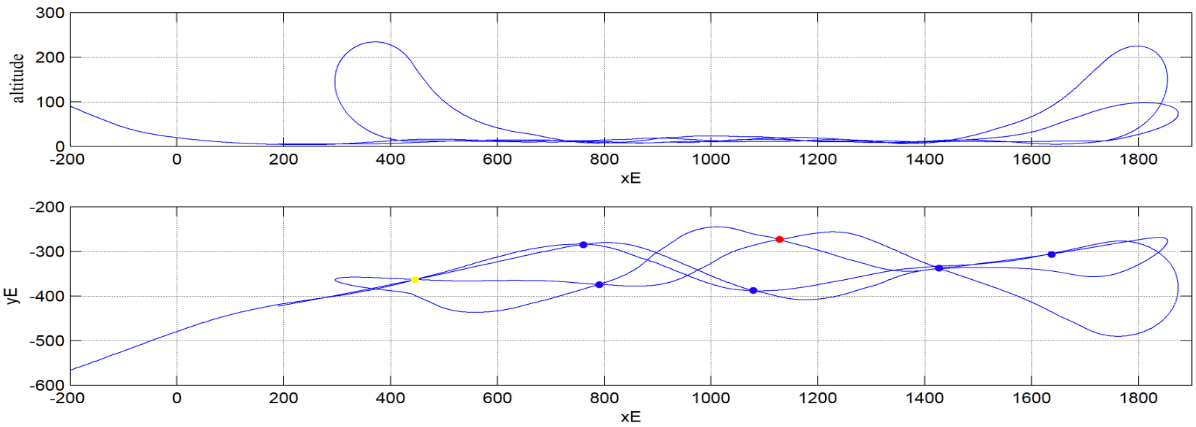

Figure 2: Flight path of simulated aircraft along a race track. The coloured dots represent the gates in race track of Red Bull Air Race Budapest 2008.

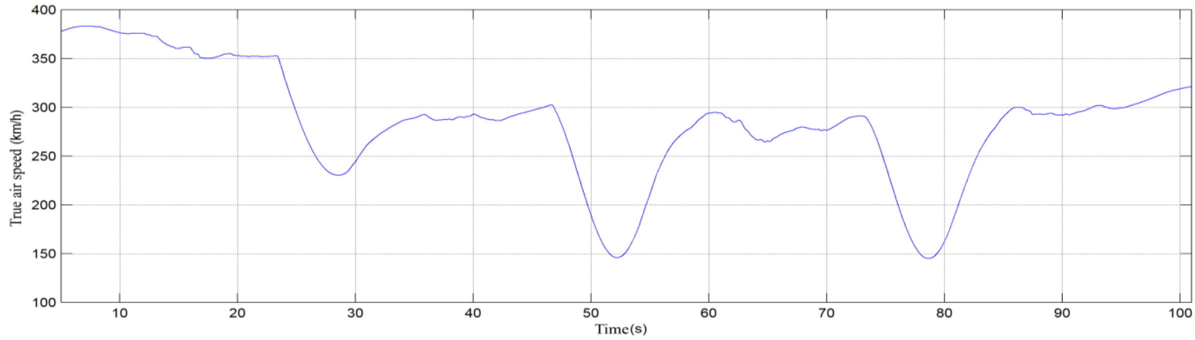

Figure 3: True air speed varies with time as airplane flies along the race track

MATLAB simulation was also used to model the longitudinal motion of a paraglider to identify the characteristics of a real paraglider [29.].

MATLAB and Simulink give opportunity to design automatic control for aircraft. Students can develop their own control algorithms to control an aircraft. They can test it, and make it perfect in the simulator before they upload their own algorithm to any UAV aircraft of the department, and in this way they can prevent accidents in real flight tests.

For students, having more knowledge in programing, there is another way to develop their skills. A new student project was started, where they are developing a stand-alone flight simulator in high level programing languages such as Delphi, C/C++, etc. Microsoft's Flight Simulator X (FSX), and the Lockheed Martin's Prepar3D [10.]. Both use Microsoft ESP technology, with Simconnect module give more opportunities to develop modern simulation environments. The Simconnect module makes possible, better then FSUIPC module in older FS versions, to use the simulator just as a graphical representation of simulated objects, where objects can be hot air balloon, airplane, helicopter, spacecraft, ship, submarine or just a car. Using these modules, FUIPC or Simconnect, makes possible for students to develop and test their own instruments and safety systems in the simulator. They can develop systems like Honeywell's Runway Awareness and Advisory System (RAAS), air navigation software for PDA, or any mobile devices, such as smart phones, tablets.

4. Recent developments in Flight Simulator laboratory

The department has a fixed base, Boeing 737NG flight simulator. It was built in 2002 for demonstration, research, education and development purposes. In 2010 the whole system was modernized and a new system was born. Staff members of department, PhD, MSc and BSc students worked together in this project. A new hardware system was developed. The students participated in this work, learnt how to design printed circuit boards using EAGLE 5.4 software [25.], how to write programs to microcontrollers using Mikropascal PRO software [25.], and how to communicate on different ways with the computer, which is running simulator software, such as RS-232, Can bus, etc.

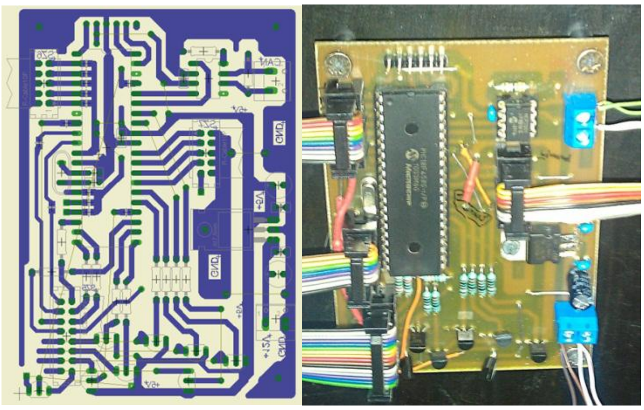

Figure 4: one of the PCBs of MCP panel, developed by students.

About 20 different hardware modules (navigation radios, autobrake panel, clock, EFIS panels, modules to MCP panel, PCBs of controls, landing gear lever, etc.) were built by students during the modernization, which finally cover almost the whole instrument panel of the cockpit. The building, programing and testing process of hardware were done in the digital electronic laboratory of department.

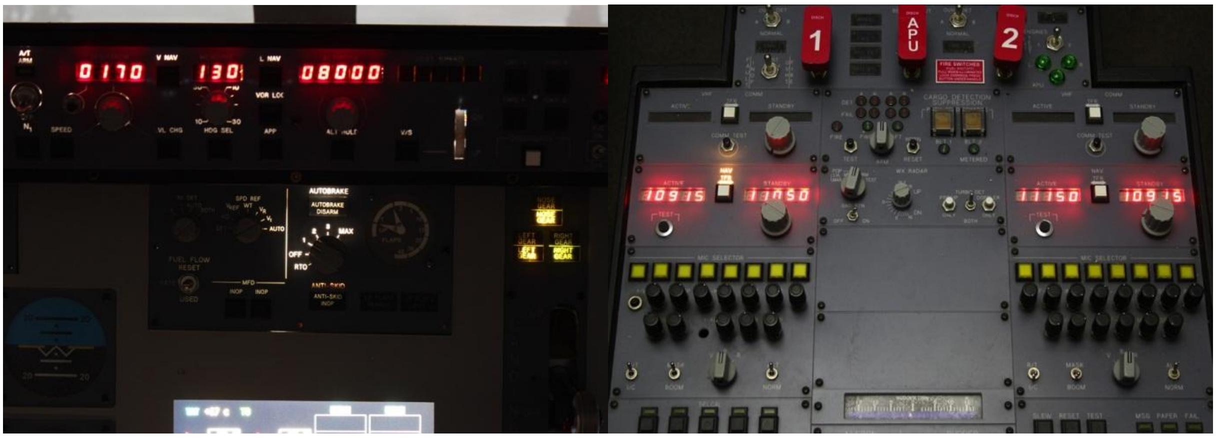

Figure 5: The autobrake panel, with MCP panel, and the navigation radios.

Students developed interface software to the new system. This software enables the developed hardware system to communicate via CAN bus with the applied simulation software and vice versa. This, similar with other software used in the simulator laboratory, was developed in Delphi7 environment, because the students during their studies compulsorily learn PASCAL programing language and Delphi as object oriented programing language. A GPS receiver simulation program, and data acquisition software were also developed. The GPS receiver simulator can be used as a mock up GPS connected to navigation devices. The Data Acquisition software can be used in test performed in the simulator, or in Flight Measurement facultative subject, where students learn about flight test engineering, how to perform flight tests and they can practice how to organize it, do it, and analyse its results in a virtual world without any real safety or financial concerns.

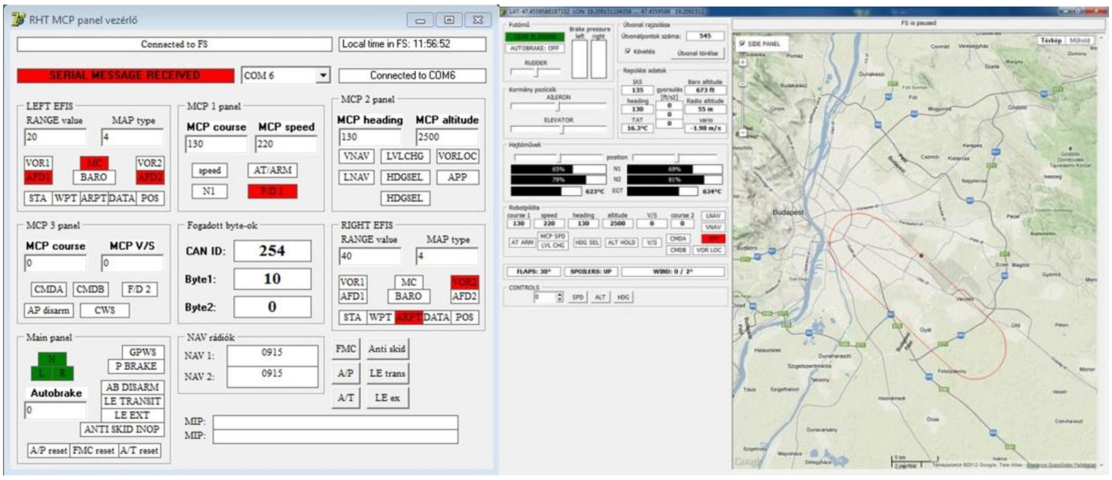

Figure 6: The GUI of interface software and the software used in demonstration flights providing general flight information for the audience.

During this work students could develop different design skills, by identifying and solving many problems, they have never faced before. With the practice and knowledge achieved in this work, they can develop and design complex measurement and data acquisition systems alone for different purposes. As a result of developed skills, one of the students developed a control stick with integrated sensors for the simulator in his diploma thesis. [31.] His system can measure and log the heart rate, heart rate variability, skin resistance, grasp force, and body temperature. During the development process, he designed a comfortable hilt which holds the user’s hand, and contains all of the sensors in suitable place. To form the best shape of hilt he used modelling paste, which was scanned into 3D model by using David Laser scanning technology. [26.] Since this work, a human sized laser scanning laboratory was built in the Department with the help of students to scan much bigger objects such as UAVs, propeller blades, paraglider harnesses, etc. to do CFD, FEM investigations in ANSYS.

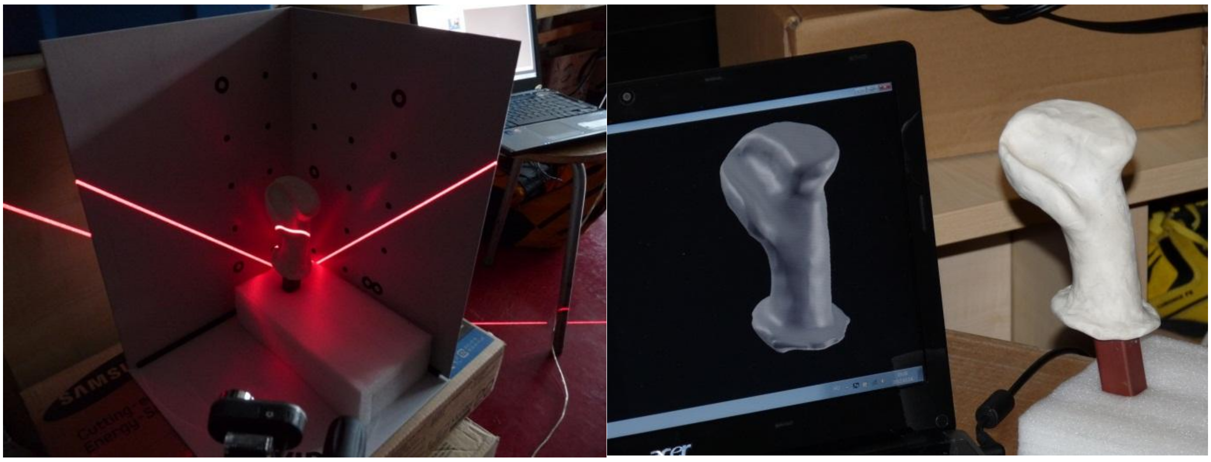

Figure 7: Laser scanning of hilt, and its 3D model.

The model, resulted from laser scanning, was corrected, and the further design processes (including the gimbal mechanism of stick) were done in CATIA V5 system. [27.]

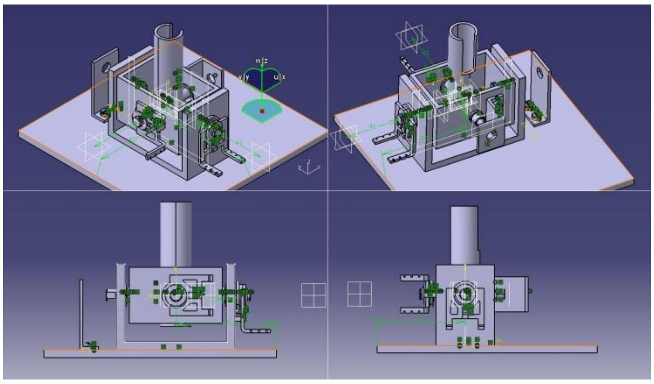

Figure 8: Gimbal mechanism of the sidestick developed in Catia

The student developed the required PCBs of the measurement system, and programmed the microcontrollers. These PCBs send data from sensors to computer via USB connection, where the developed software processes, and represents the measured values and logs them for further investigations. The developed stick with sensors works perfectly in flight simulator laboratory.

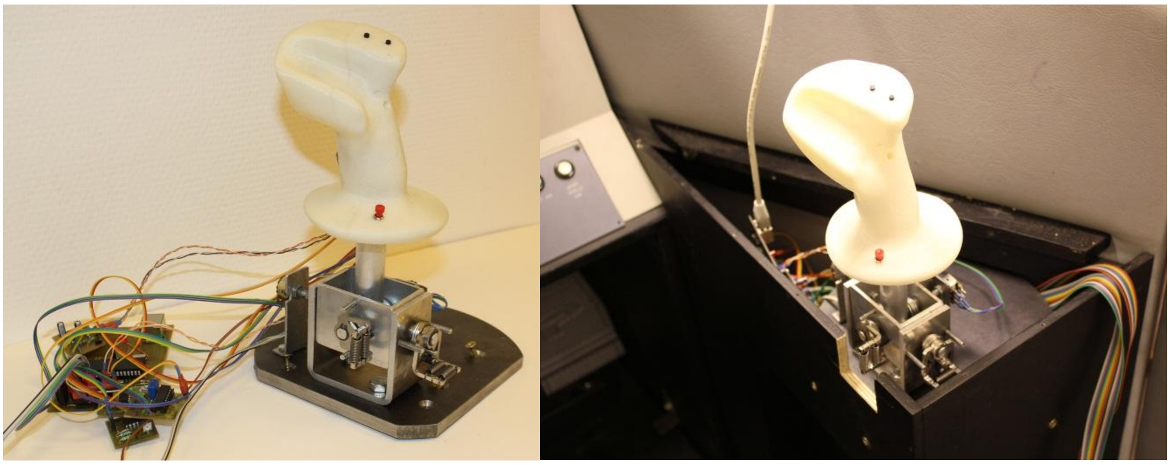

Figure 9: Prototype of developed stick with integrated sensors to measure pilot’s physiological condition

As a continuation of this work, other student in his diploma thesis continue to develop this system, by solving the teething problems emerged during the tests, and develop another one, to do some real life measurements in gliders.

5. Fluid flow modelling with ANSYS Workbench

The evolution of computational performance have made possible to the Finite Volume Methods to become widespread. The finite volume method is a discretization method which is well suited for the numerical simulation of various types (elliptic, parabolic or hyperbolic, for instance) of conservation laws; it has been extensively used in several engineering fields, such as fluid mechanics, heat and mass transfer or petroleum engineering. [13.] The field fluid dynamics, so-called Computational Fluid Dynamics (CFD), has got particular featured role in fluid dynamics physics and aircraft engineering education, and in the last decade it has showed its advantages. Because of tremendous wealth of information available from numerical simulation, certain fundamental concepts can be efficiently communicated using an interactive graphical interrogation of the appropriate numerical simulation data base. In other situations, a large amount of aerodynamic information can be communicated to the student by interactive of simple CFD tools on a workstation or even in a personal computer environment. [14.]

The up-to-date education of CFD methodologies and applications takes very important role in the activities of BME Department of Aircraft and Ships. For today's undergraduate aircraft engineering students it is indispensable to properly know one CFD software at least. Deploying software from ANSYS, Inc. as an integral part of the engineering curriculum allows our Department to train the future work force with world-class simulation tools and technology and familiarize them with Simulation Driven Product Development processes. [15.]

With the CFD simulations rich visualization capabilities can be realized, which are extremely useful in developing a physical understanding of fluid flows. The results can be illustrated in plots, coloured figures and solids or surfaces, with which the demonstration of axisymmetric temperature field in a pipe or the velocity field and pressure distribution in the environment of a wing are more understandable. An assumption in the experiment can be confirmed when students can see that the flow is well mixed at the end of the adiabatic mixing section. The comparison between experiment and simulation for the temperature variation along the pipe axis can be plotted along with results from the original and refined meshes. In that way the validation process of a created model can be delineated.

In the profile of the Department the educated simulations are focused on understanding the behaviour of liquids and gases, which are crucial for engineers who need to predict and improve the performance of new designs or processes. In the course of Fluid- and Thermo Machines in Vehicles the basics of Ansys Workbench are taught, but deeper knowledge can be achieved with participating in the Scientific Conference of Students events. In the last years, several students prepared with some kind of simulation, which was modelled in Ansys environment. Some of the students with more experience and knowledge are ready to get involved into industrial and research projects, where besides the single-phase flow simulations the application of ANSYS is extended to further simulations, where besides the fluid flow, other physical processes are be modelled, like combustion modelling in a gas turbine combustion chamber.

This latter project was developed by a student in his diploma thesis. The goal was performed about the modelling of heat transfer processes in a gas turbine combustor. The base of the simulation was the Solar Turbine's Taurus T60 industrial gas turbine. The CAD model of the combustor was created in Catia environment and simplified where it was reasonably possible to reduce the computational time consumption. Because not all boundary conditions were known, a concentrated parameter distribution calculation was completed to determine missing parameters for boundary conditions. k-ε turbulence, Finite Rate Chemistry/Eddy Dissipation Model combustion model and P 1 radiation model were used in the calculations. The results of the simulation were compared to the measured data at the nominal operation to have information about the correctness of the analysis. The model was extended with the liner, secondary flow and housing in order to investigate the effect of polytrophic process on heat transfer and the new results were also compared to the measured data.

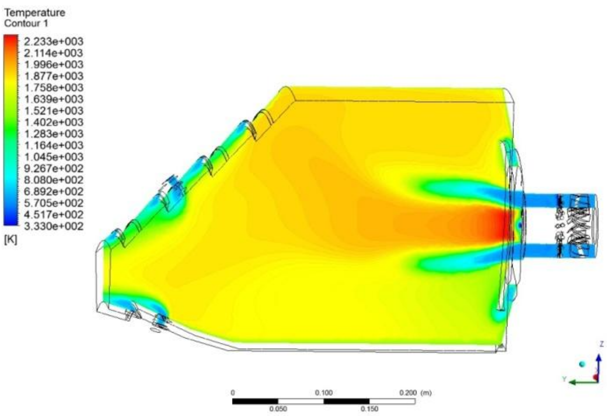

Figure 10: Temperature distribution in the mid-section of combustion chamber

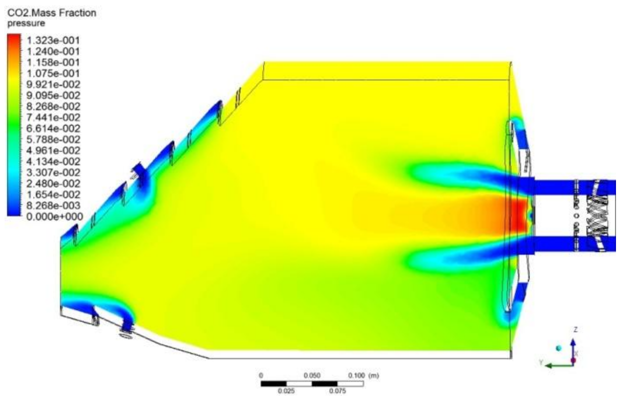

Figure 11: CO_2 mass fraction in the mid-section of the combustion chamber

Another, interesting industrial project was an acrobatic airplane development project. The goal was to improve the flight parameters of the Corvus Racer 540 high performance acrobatic aircraft in manoeuvres, which have got primary importance in the Red Bull Air Race Competition.

The goals of the simulation were the following:

- Analysis of behaviour of acrobat aircraft during the curve manoeuvre,
- 3D CFD Analysis of rotating motion of wing (with Eppler 472 profile),
- Investigation of aileron effect by opened and closed state,
- Calculation of lift force and drag force by maximum angle velocity:
  3D CFD Analysis for comparison of three different wing type (Eppler 472; Joukowsky acrobat; NACA 0012),
- Investigation of possible modifications on section contour for improved design specifications (C_L-α, C_D-α and Polar curves),
- CFD (Computational Fluid Dynamics) simulation and analysis of different concepts,
- Importance of the numerical analysis of the complete 3D aircraft,

As the boundary conditions were well-known, with the simulations the results were satisfying, and based on them, significant changes were achieved. These upgrades were implemented in the later configuration of the aircraft and have improved the flight characteristics of the aircraft.

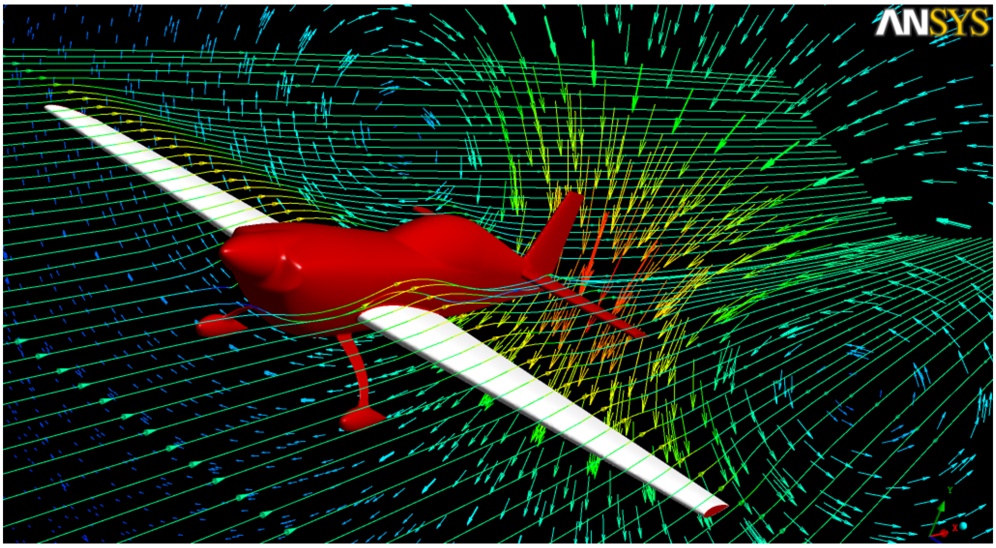

Figure 12: Velocity vectors behind the aircraft projected in the cut surface and coloured by velocity at α=10

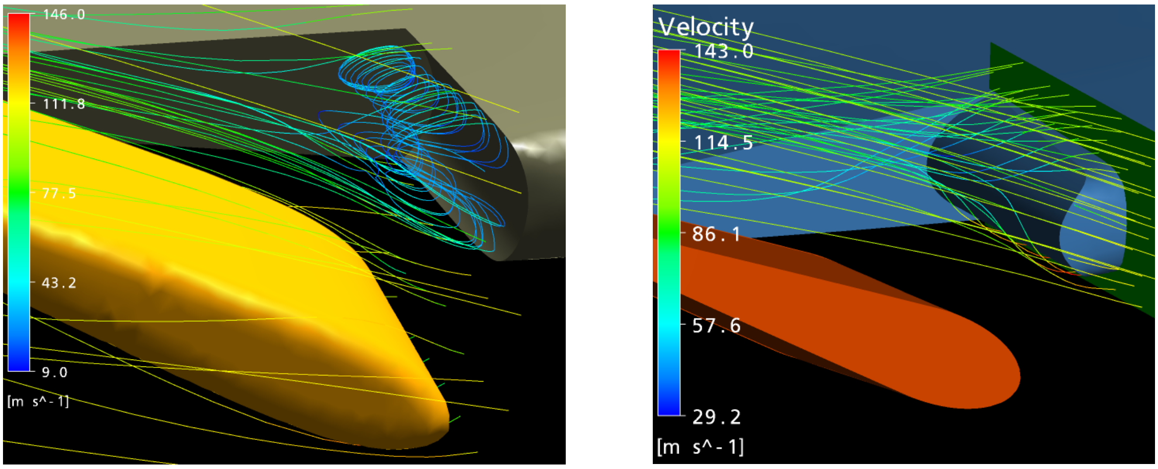

Figure 13: Streamlines between the wing and the aileron (aileron deflection) in the case of old wing configuration and in the case of the new wing configuration

6. Own-developed fluid flow modelling: DASFlow

Student with high knowledge and experience or PhD students participate in the development and application of the Department's own CFD software, called DASFlow. With these work the fundamentals of numerical approximations (like Newton Rapson method) and solvers can be learned by understanding the equation solving methods. The high knowledge of conservative equations, their discretization methods (like Runge-Kutta method) and other high level of mathematical cognitions are learned in the end of the studies. But with these experiences, new methods and solutions can applied, like the usage of invers design methods. In a research project, a two dimensional inviscid compressible flow solver has been implemented, validated and extended for completing inverse design based optimization problems. Euler equations are used for modelling basic physics. The standard cell centred finite volume method has been applied with Roe's approximated Riemann solver, MUSCL (Monotone Upstream Schemes for Conservation Laws) approach and Mulder limiter. Stratford's separation prediction method with constrained SQP (Sequential Quadratic Programming) optimization is implemented to evaluate the optimum pressure distribution at strong adverse pressure gradient flow conditions. A particular inverse design method is developed to complete wall surface modification till the optimum target pressure distribution is reached by means of inverse, wall modification and direct algorithms. The direct solver is validated and the coupled system of the inverse design method with constraint optimization is tested over the NACA 65-410 airfoil. [17.]

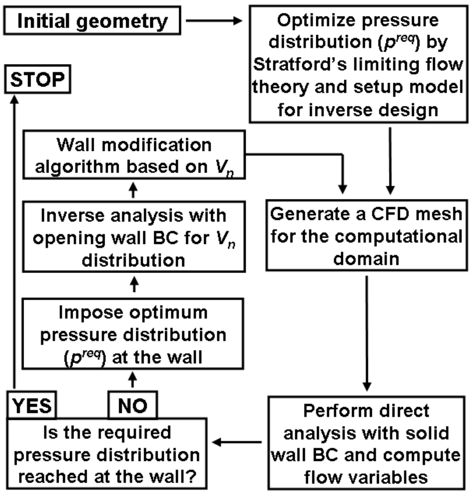

Figure 14: Flowchart of the computational procedure of the iterative inverse design calculation

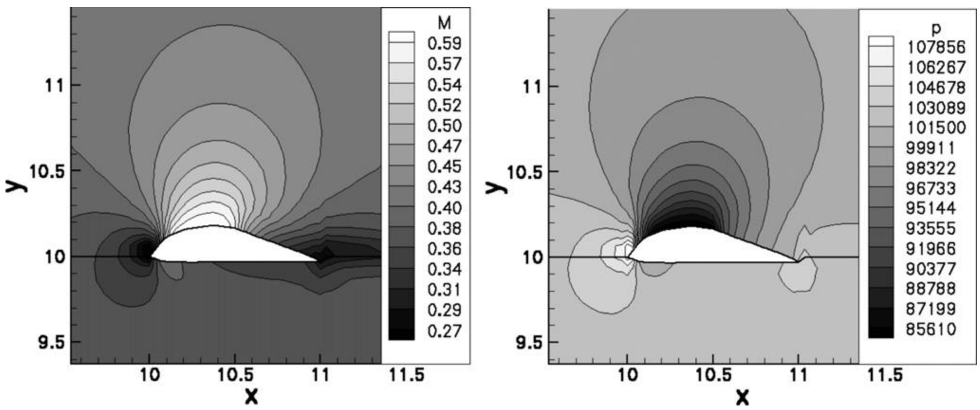

Figure 15: Mach number and pressure distribution of the result case (result of the inverse design based optimization procedure)

7. Automation and monitor of measurements with LabVIEW

The LabVIEW is a platform-independent, powerful, graphical programming development environment, which can be used for data acquisition, storage, analysis and presentation. Its advantage, that it fulfils industry standards and can be easily applied for measurement and automation. Thus, for engineering students it is easy-to-use software support to plan and to perform a physical measurement with real and virtual instrumentation.

Engineering students learn a wide scale of physics during their education. Recently, the demonstration and illustration of the different laws became easily applicable, and had become a vital part of the education. Despite many other academic fields, in physic it is possible to reproducibly illustrate the precise and deterministic character of how nature behaves and structures the universe. To understand these processes, real-world demonstration experiments can be demonstrated as proofs that the main concepts, communicated verbally, are valid in the real world. [17.]

With LabVIEW could be helpful during these physical experiments, because makes possible quick implementation of measurements during demonstration experiments and the reduced need to build dedicated hardware. Additionally, the modification is easy, the access and practical storage on a PC storage driver allow easy usage even for a strongly diversified inventory of demonstration equipment. [17.]

LabVIEW uses so-called virtual instruments (VIs). They're programs, which appearance and operation imitate physical instruments. These VIs have got a front panel which is the interface of the user, and a block diagram, which is considered as a LabVIEW program and a connector pane which is a set of terminals on the icon that corresponds to the controls and indicators of a VI, similar to the parameter list of a function call in text-based programming languages. [18.]

The inputs – such as push buttons, sliders, switches or values entered by user from the keyboard…) and the outputs (like indicators, graphs, LEDs, charts…) can be inserted in the front panel. The front panel can be created from the controls palette containments. Block diagrams integrate terminals, subVIs (subVIs are similar to a function in a text-based programming language), functions, constants, structures, and wires, which transfer data among other block diagram objects. Nodes are objects on the block diagram that have inputs and/or outputs and perform operations when a VI runs. [18.] The nodes are functions, structures, or subVIs. The block diagram can be created from the function palette containments (functions and constants).

Besides the most common execution structures (While loops, For loops, Case structures), LabVIEW offers more, advanced types, like Shift registers, Tunnels, Arrays or Event Structures. A LabVIEW programs follow a dataflow model for running VIs. When all the required inputs are received of a block diagram node, it will be executed. [18.]

With these tools, the automation of measure processes, the data collecting and digitalizing are much easier, and the main principles can be taught during labour lessons. For the applications two examples are shown in this article. The first application is a sound-based tachometer, designed by a student. The measurement's advantage against other methods, that the requirements are only a laptop with microphone or line input with LabVIEW development environment and Sound & Vibration Toolkit installed on it and a multimedia microphone. The monitoring tool (microphone) doesn't have to be in contact with the measured equipment (for example a gas turbine).

The program convert the input signs into digital form, and with Fast Fourier Transformation (FFT) creates the Power Spectrum of the received sounds in every second. In the Power Spectrum, the periodic components, like the rotation caused frequency, appear in peaks. If the necessary digital filters and other additional subVI-s (such as an optional frequency range option) are installed correctly, the program can search the most characteristic peak, which shows the rotational speed of the monitored equipment. (Figure Figure 16and Figure 17 shows the program's earlier stage).

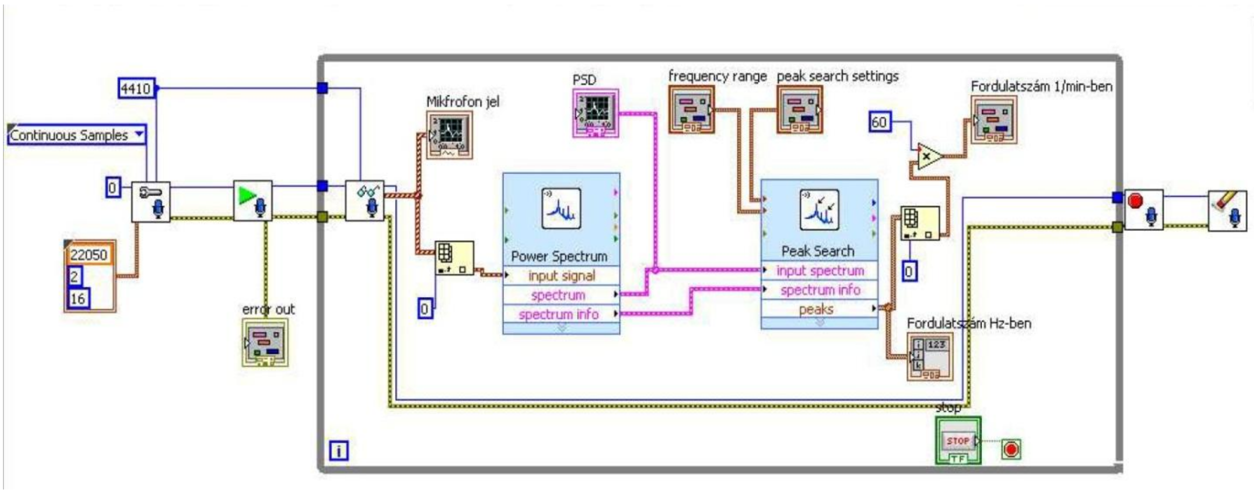

Figure 16: The Block Diagram of the tachometer in LabVIEW

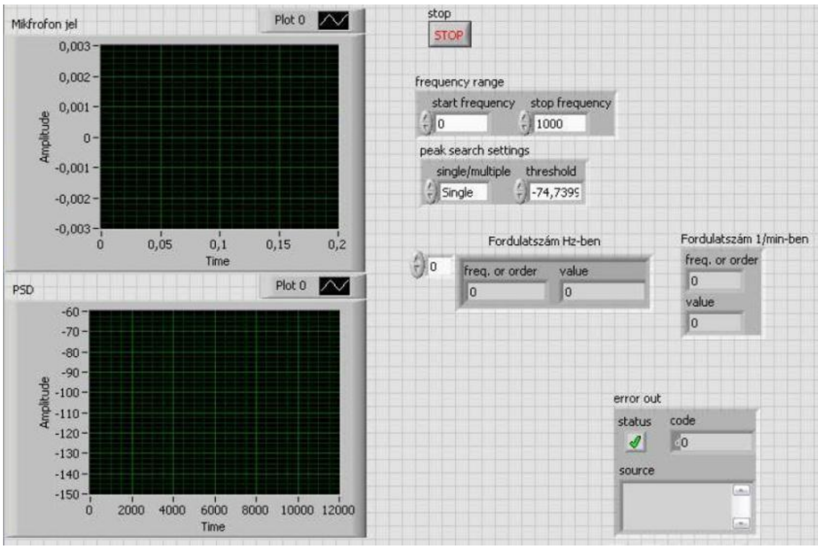

Figure 17: The Control and Monitoring panel of the tachometer in Labview

The other application is a lot more complicated. In this case, a small gas turbine's regulator and data collector and acquisition system was developed. The gas turbine originally was a TSz-21 starter gas turbine of MiG-23 or Szu-22 aircrafts, which was modified to an experimental jet engine, which is marked as TKT-1. [20.]

The measuring system processes the most relevant data for the users: temperatures (before and after the compressor, after the turbine, in the lubrication oil tank, on the electric motor of the oil pumps); pressures (environmental, before and after the compressor, after the turbine); mass flow; engine rotary speed; thrust; gas-velocity control lever position indicator. [20.]

The appropriately designed computer display makes possible to monitor more data at the same time, and allows storing the measured values, for later experiments and analysis. [20.] The program is developed continuously, and several students are involved in these investigations. Furthermore, during a whole course, aircraft engineering students can apply their own measurements with these tools.

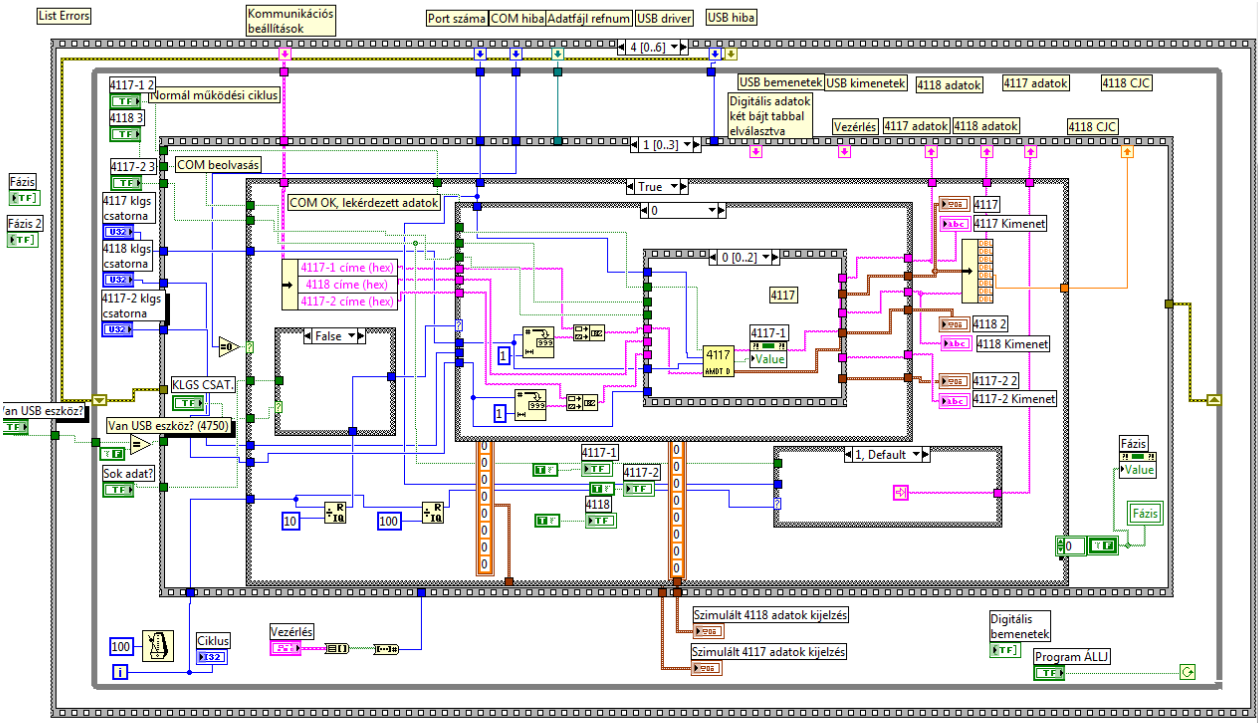

Figure 18: The Block Diagram of the TKT-1 gas turbine regulator and monitoring system

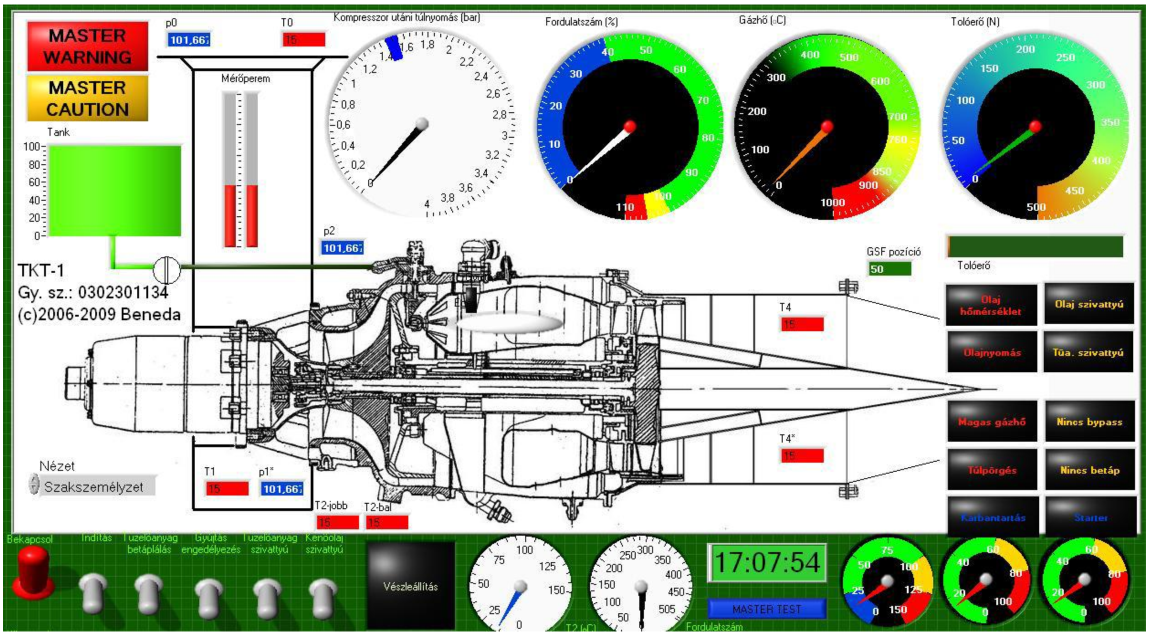

Figure 19: The Control and Monitoring surface of the TKT-1 gas turbine regulator and monitoring system

Conclusions

The article is about high level engineering software support of aircraft and design skill development. After the short overview of the technology level nowadays, the software teaching method of the BME Department of Aircraft and Ships was introduced. In this method, the education of engineering software are integrated into the professional courses, in that way the practice, software using skills and theory can be taught at the same time, and the relationship between them can be pointed at directly. Students with more interest are got involved into industrial development and research projects, thus they can show their best, get a lot of new experiences and upgrade their knowledge to higher level.

The Department has got several licenses for high level engineering software. The second part of the article introduces these software, and their usage in education. Also some interesting industrial development and research project, where BSc, MSc and PhD students were involved, and developed some useful industrial applications.

In Aerodynamics, Flight Mechanics, Flight Measurements, etc. subjects the usage of high level software is high. Using different software, students can deeply understand and learn the theory of flight in a practical way. Students having interest in micro controllers, programing, or just being eager to use their knowledge were involved into different projects, to develop software, hardware or complete measurement and data acquisition systems in which they could develop further their skills in design, research and development.

For CFD analysis – sometimes completed by other physics, like combustion – and FEM analysis Ansys Workbench software is applied. To validate the simulated values, it is necessary to do some experimental measurements, which can be supported by LabVIEW, to monitor, collect and acquisite data, furthermore the measurement can be controlled with this software. Students with higher mathematical, physical knowledge and more experiment can develop the Department's own CFD software, which for example can be used for inverse design methods.

References

[1.] W. Pachatz, A. Zafoschnig: Education Standards and Competence-oriented Curricula – The Austrian Technical Colleges Take a New Approach to Excellence; iJEP – Volume 1, Issue 3, October 2011; http://dx.doi.org/ijep.v1i3.1818

[2.] A. Aan, M. Heinloo, E. Aarend: Interactive Computer Aided Learning and Teaching of Analytical Mechanics; iJEP – Volume 1, Issue 3, October 2011; http://dx.doi.org/ijep.v1i3.1788

[3.] F. Jorge Lino, Teresa P. Duarte: Research Skills Enhancement in Future Mechanical Engineers; iJEP – Volume 1, Issue 1, April 2011; doi:10.3991/ijep.v1i1.1590

[4.] José M. M. Couto Marques: Experimental, Numerical and Virtual Tools in Civil Engineering; iJEP – Volume 1, Issue 1, April 2011; doi:10.3991/ijep.v1i1.1605

[5.] A. Zeidmane, S.Cernajeva: Interdisciplinary Approach in Engineering Education; iJEP – Volume 1, Issue 1, April 2011; doi:10.3991/ijep.v1i1.1604

[6.] A.J. Sierra, T. Ariza, F.J. Fernández, G. Madinabeitia: Tool for Validation Software Projects in Programming Labs; iJEP – Volume 2, Issue 2, April 2012; http://dx.doi.org/10.3991/ijep.v2i2.2080

[7.] A. G. Abulrub, A. Attridge, M. A. Williams: Virtual Reality in Engineering Education: The Future of Creative Learning; iJET Vol 6, No 4 (2011); http://dx.doi.org/10.3991/ijet.v6i4.1766

[8.] Maura Borrego, Jefrey E. Froyd, T. Simin Hall: Diffusion of Engineering Education Innovations: A Survey of Awareness and Adoption Rates in U.S. Engineering Departments; Journal of Engineering Education, July 2010

[9.] V. Thurner, A. Böttcher and V. Winter: Soft Skill Development along the Education Path - Evaluating expectations on and perceptions of student competencies in Software Engineering education; iJEP – Volume 2, Issue 3, July 2012; http://dx.doi.org/10.3991/ijep.v2i3.2148

[10.] Tiia Rüütmann, Hants Kipper: Teaching Strategies for Direct and Indirect Instruction in Teaching Engineering; iJEP – Volume 1, Issue 3, October 2011; http://dx.doi.org/ijep.v1i3.1805

[11.] M. Ivanović, S. Xinogalos, Ž. Komlenov: Usage of Technology Enhanced Educational Tools for Delivering Programming Courses; iJET – Volume 6, Issue 4, December 2011; http://dx.doi.org/10.3991/ijet.v6i4.1796
[12.] I. Handlovská , T. Kozík: The Reduction of Interest among Elementary Students in the Field of Technical Education; iJEP – Volume 1, Issue 3, October 2011; http://dx.doi.org/ijep.v1i3.1822
[13.] Robert Eymard, Thierry Gallou¨et, Rapha`ele Herbin: Finite Volume Methods; Handbook of Numerical Analysis, P.G. Ciarlet, J.L. Lions eds, vol 7, pp 713-1020, October 2006
[14.] Terry L. Holst: Computational Fluid Dynamics Uses in Fluid Dynamics/Aerodynamics Education; NASA Technical Memorandum 108834, July 1994; Ames Research Center, Moffett Field, California 94035-1000; http://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/19950004435_1995104435.pdf
[15.] Ansys CFX: Improving the Speed and Accuracy of Engineering Simulation with Windows® HPC http://www.ansys.com/staticassets/ANSYS/staticassets/partner/Windows7/ansys-cfx-windows-hpc-server.pdf
[16.] Rajesh Bhaskaran: Teaching Simulation to Future Engineers, ANSYS Workbench facilitates integration of simulation into the engineering curriculum, ANSYS Advantage • Volume IV, Issue 1, 2010; http://www.ansys.com/staticassets/ANSYS/staticassets/resourcelibrary/article/AA-V4-I1-Teaching-Simulation-to-Future-Engineers.pdf
[17.] Árpád Veress , Attila Felföldi, Tamás Gausz, László Palkovics: Coupled problem of the inverse design and constraint optimization; Applied Mathematics and Computation; September 2011; doi:10.1016/j.amc.2011.08.110; http://www.lauterburg.ch/UrsLPublications/LV-PhysicsWPPrint.pdf
[18.] Olfa Boubaker: National Instruments® LabVIEW™: Ultimate Software for Engineering Education; National Institute of Applied Sciences and Technology, Tunis, Tunisia; http://cerc.wvu.edu/download/WORLDCOMP%2711/2011%20CD%20papers/FEC4450.pdf
[19.] K. Beneda: TKT-1 Maintenance Handbook; http://rht.bme.hu/letoltesek/Beneda%20Karoly%20anyagai/TKT-1/%C3%9Czembentart%C3%A1si%20k%C3%A9zik%C3%B6nyv/70__TKT1_1329145.pdf
[20.] K. Beneda: Teljes Hatáskörű Digitális Gázturbina Szabályzás (FADEC1) Fejlesztése Kisméretű Sugárhajtóműves Berendezéshez; Repüléstudományi Konferencia, April. 2009. http://www.szrfk.hu/rtk/kulonszamok/2009_cikkek/Beneda_Karoly_Tamas.pdf
[21.] Aircraft Design Software: http://www.pca2000.com/en/index.php
[22.] Advanced Aircraft Analisys: http://www.darcorp.com/Software/AAA/
[23.] MATLAB, Simulink: http://www.mathworks.com/products/matlab/
[24.] EAGLE: http://www.cadsoftusa.com/eagle-pcb-design-software/
[25.] Mikropascal PRO: http://www.mikroe.com/mikropascal/pic/
[26.] David Laserscenner: http://www.david-laserscanner.com/
[27.] Dassault System CATIA: http://www.3ds.com/products/catia/portfolio/catia-v5/latest-release/
[28.] J.Rohacs, A.Veress, I.Jankovics, A.Voloscsuk, Cs.Farkas: Goal Oriented Aerodynamic Design of a New Acrobatic Aircraft, READ2010 June 28-30. Warsaw, ISSN 1425-2104
[29.] A.Nagy, J.Rohacs: MEASURING AND MODELLING THE LONGITUDINAL MOTION OF PARAGLIDERS (ref.num.:832), ICAS 2012, Brisbane, Australia
[30.] Lockheed Martin Prepar3D: http://www.prepar3d.com/
[31.] A. Szabó: Pilóták fiziológiai jellemzőinek repülés közbeni mérésére alkalmas kormányszerv fejlesztése, Diploma Thesis (in hungarian), 2012