

Nr. crt.	Title	Description	Requirements	Tests	hours / day
1	Electronic Simulation Library build up	<p>The continuous growth of the electronics circuit and systems requires increasing the simulation and verification activities during the entire engineering development cycle. In order to support the simulation activities, a central library with own simulation models is required to be developed based on customer's specifications.</p> <p>The main goals of this project are:</p> <ul style="list-style-type: none"> - Collect the specifications from a customer and build up simulation models for specific components and store them into a central library - Document the methodology for model creation, test and validation and share them across the central support group. 	<p>Electronic fundamentals (BJT, MOS-FET, Diode, ...)</p> <p>Electronic circuits and systems, modeling and simulation</p>	IT, HW	4, 6, 8
2	Iterative Learning Control: Convergence of Adaptive Learning of Injector Needle Closing Point for a PCR Direct Drive injector	<p>Identify an analytical method to determine the convergence robustness for the Iterative Learning Control of injector needle closing point for a Piezo Common Rail Direct Drive Injector . Design For Six Sigma (DFSS) method will be used.</p> <p>Develop a Matlab application to determine the robustness of the existing adaptive algorithm for a large array of injector working points (high/low energy, high/low temperature, multiple injections pattern).</p> <p>The HiL test bench will be available for experiments. A thermal injector box will be used for thermal experiments.</p>	<ul style="list-style-type: none"> - Systems Theory: good - Systems Identification: good - Numerical Control Systems: good - Signal Processing: good - Optimization Techniques: good - Programming ANSI C: good - Electronics : good 	ANSI C, microcontrollers, Matlab	6

Nr.c rt.	Title	Description	Requirements	Tests	hours / day
3	Auto Code Generation ACG from Matlab Simulink models for customer projects	<p>We receive from customer models in Matlab Simulink format. We need to generate code out of these models and tests.</p> <p>In order to reach that we need to follow several steps:</p> <ul style="list-style-type: none"> - generate test vectors using Reactis from customer models - import customer models and modify in SDA format (Continental specific Matlab library) - generate specifications - auto code generation, review generated code, solve error/warning messages - import test vectors and run test cases in floating point, fixed point, software in the loop (SIL) and on the target with microcontroller (PIL), analyze test results - document, save results in a MKS workspace 	<ul style="list-style-type: none"> - Systems Theory: good - Systems Identification: good - Numerical Control Systems: good - Signal Processing: good - Programming ANSI C: good 	ANSI C, Matlab	6, 8
4	Automatic testing of an Airbag Control Unit	<p>As a student in Passive Safety Test Group you will learn about Airbag Control Unit, get in contact with the latest test equipment on the market, use SW functions to control complex/Real Time measuring equipment.</p> <p>Inside our team we have several exciting projects open for you:</p> <ul style="list-style-type: none"> - develop HW & SW tools that enable automatic test activities for the Airbag Control Unit. - develop automatic tests using C# and the preexisting Framework. - maintain & Develop Excel macros for test result evaluation 	<p>Good programming skills.</p> <p>Basic electronics & uController know-how</p>	C++, C#, HW	6, 8

Nr.c rt.	Title	Description	Requirements	Tests	hours / day
5	BLDC and DC motor performance testing	The project consists in studying and understanding of project specification relative to the TCU (Transmission Control Unit) electrical tests, BLDC and DC motor functionality and TCU functionality.	Electronic components, microcontrollers	Microcontrollers, HW	6
6	Optimization of functional elements in mechanical design of ECU17 platform	To get accustomed with new requirements of ECU17 mechanical platform To get accustomed with design guidelines and internal mechanical standards To be integrated in the extended project team To create 3D models and drawings for the defined functional elements of ECU17 mechanical platform To validate the solutions	CAD (preferably ProEngineer) Machine elements Tolerance calculation Technical drawing Mechanic technologies	Mechanics	6
7	Wireless Test Tool for Vehicle Key ID	The Wireless Test Tool is a PC application written in Visual C#. The application implements a user interface that controls a device (attached to the PC via USB) to perform the following tasks: - Send Wireless data to Vehicle Key ID - Perform button presses of Vehicle Key ID - Control the supply of Vehicle Key ID The purpose of this thesis is to Identify together with the coach improvements/new feature which can be added to the wireless test tool, and implement them.	C, microcontrollers	C, microcontrollers, Visual C#	6

Nr.c rt.	Title	Description	Requirements	Tests	hours / day
8	Near Field Communication Configuration Tool	Develop a tool used to configure the behavior of a system which used Near Field Communication (NFC) technology for information exchange the car and NFC capable devices (tags, smartphones) The summer practice project implies: - the creation of an MS Windows application - used to configure the system, and which will generate C source files - testing the generated source files on embedded hardware	Visual C#, C, microcontrollers	C, microcontrollers	6
9	Compare tool for RF DRV configuration files	Design of a tool in a high level programming language (at free choice) which will compare the values of RF DRV registries (TDA5235, TDA5240) from a *.txt input file against the values from a C source file (arcfg_drv52xx.c); Design of a tool in a high level programming language (at free choice) which will create a trace log (in a .txt file) of events thrown by RF Test SW. The tool will interpret the sequence of hex values and add a literal comment near/or on the same line to make the debug log easy to read and comprehend; Redesign or improvement of TestLib Environment (C#); DLLs, interface with the equipments (SMIQ, Emulator Interface, etc.).	HL Programming languages - C#, C++, Java, VBA (Visual Basic) Embedded systems, electronics Strong knowledge of Microsoft Office tools (Excel, Word)	C++, C#	6
10	DB Compare Tool	Compare MS Access databases: structure and data The application requires to: - compare the structures of databases - read and compare data from databases - create a clean and straightforward user interface for showing the differences - merge the differences between databases	C#/Java, OOP, General DB knowledge, Logic, Windows User Interface (custom controls, data exchange, user interaction), Ability to read, understand and follow a basic set of requirements	C#, Java	8

Nr.c rt.	Title	Description	Requirements	Tests	hours / day
11	Instrument cluster automated testing by using a newly designed simulation box	<p>The project consists in developing automated test cases for various instrument clusters by using the newly designed generic test box which is able to simulate CAN communication, diagnose, digital switches, analogue sensors, voltage profiles, etc.</p> <p>Automated sound tests will also be performed by using an anechoic chamber, a professional sound capture setup (hi-def microphone and dedicated sound acquisition cards with real-time processing), together with in-house developed sound-recognition algorithms.</p>	Electronics, Basic Programming Skills	microcontrollers	6
12	CAN I/O Control	<p>1. Develop an embedded application to change the outputs of an ECU based on information received in a CAN message. The application has to check the received information and if it is valid then change the value of the outputs accordingly. Otherwise, it has to provide the correct diagnostic values</p> <p>2. Develop an embedded application that sends a CAN message based on the inputs of an ECU. The application has to check the input values and based on predefined values, it has to send a CAN message to one ECU or another.</p>	Basic uController knowledge, C, Analytical thinking	ANSI C, microcontrollers	8

Nr.c rt.	Title	Description	Requirements	Tests	hours / day
13	Decoder for SPI sensor communication	<p>Currently a not reusable filtering and Macros methods in Excel are used to interpret the high amount of data from ASG5 sensor. This is realized manually by developers/owners who realize/use SPI communication.</p> <p>Project consist in adaptation of another tool to be used for SPI connected devices ASG51 . The Imported SPI trace, translated to Tx Rx sequence can be analyzed by using a special CAPL edited code and a special .cfg file.</p>	SPI protocol	ANSI C	4, 6, 8

Nr.c rt.	Title	Description	Requirements	Tests	hours / day
14	Static C Code Analysis: removal of root cause vs. message suppression	<p>As part of the effort to ensure high code quality C files are analyzed by a static checker (lint) to find errors or deviations from MISRA* rules.</p> <p>In order to meet customers' "zero lint messages" requirement two alternatives stay at hand:</p> <ul style="list-style-type: none"> - correcting the code in order to have the message removed from the lint report or - simply suppressing the lint message. <p>While the second is the simpler alternative it also is the un-safest and it shall only be used if there is no way to remove the root cause which leads lint to report the message and only if by keeping the code unchanged (and consequently the tool would report the message!) the outcome of the application is under control/correct.</p> <p>The aim of the project is to set up two lists of lint messages:</p> <ul style="list-style-type: none"> - a list of lint messages which can be eliminated by removing the root cause in the code; each message in this list will be accompanied by code examples before and after the correction - a list of messages that are allowed to be suppressed, because there is no way (workaround) to get rid of the message and because suppressing the message is not only unavoidable but also safe; each message in this list will be accompanied by code examples and possible justification messages. 	C programming language	ANSI C	4, 6, 8

Nr.c rt.	Title	Description	Requirements	Tests	hours / day
15	Automatic Generation of Test Environment for MCRP Aggregate	<p>Automatic generation of test environment required for testing the MCRP functionality on the test bench/at project integration phase. The developer will be introduced to our CessarCT environment and MCRP functionality.</p> <p>The practice consists in developing a number of jet files to generate the test environment required during tests on test bench and/or at integration into projects (e.g. source *.c files containing test interfaces and variables/calibrations, DDS *.grl files to be used in order to generate a test A2L etc).</p> <p>The candidate required knowledge shall be: basic C programming skills and Java programming skills (medium knowledge would be a plus).</p>	C, Java programming language basic knowledge (medium knowledge would be a plus).	ANSI C, Java	6, 8
16	MFC innovative bracket	<p>The student will get knowledge of Continental brackets and the actual patent will serve as a guideline for building and developing the prototype bracket;</p> <p>An investigation will then be carried out to see how we can build up a prototype bracket to mount the camera using wire spring. A sourcing and a feasible design for the wire spring will be also conducted.</p>	Drawing knowledge, plastic part design, tolerances and Catia know-how	Mechanics	4, 6

Nr. crt.	Title	Description	Requirements	Tests	hours / day
17	SRR Bracket Smart	<p>The student will get knowledge of Continental brackets, customer requirements and SRR mounting guide and functionality.</p> <p>An investigation will then be carried out to see how we can optimize the daily work process for the SRR brackets design. For this, Power Copy features will be implemented for different bracket interfaces.</p>	Drawing knowledge, plastic part design, tolerances and Catia know-how	Mechanics	4, 6
18	Test Automation for OVIP Project	<p>The IIC Business Unit provides software solutions for Multimedia Systems. The software solutions consist of SW packages integrated together in the system. The programming language used for writing these SW packages is C++.</p> <p>The student shall develop Automatic Test Cases using our internal ATP platform (Automatic Test Platform) and based on the existing manual Test Cases. The student will have the support offered by a mentor during the project.</p>	SW Programming (any language - basic knowledge), logical thinking	C, C++	6

Nr.c rt.	Title	Description	Requirements	Tests	hours / day
19	On Board Unit Wireless	<p>The main scope of the project is to design a windows application using the .Net Framework which will be used to test a On Board Unit which has a Bluetooth interface. This windows application shall send and receive messages to the OBU. The data inside the messages represents integration tests data injection. The application shall be written in a way in which automatic tests can be written and run for a long time. When the application finishes the tests it will generate a report xml which states the tests which were performed.</p> <p>The subjects covered during this project are: Bluetooth Communication, Integration Tests, .NET Framework, XML generation, and Automatic test execution.</p> <p>Also the student shall gain knowledge about may embedded software modules which are inside the OBU thus gaining knowledge in the following subjects: GPS, GSM communication, Accelerometer, Gyroscope, Vehicle Diagnosis, Data encryption and Compression</p>	C# or C++ or Java, Basic .NET, Basic Embedded Systems, ANSI C.	ANSI C; microcontrollers; C++ or C# or Java	8
20	Small Engine(1 cylinder	<p>The base for this project is the Evaluation board for MC33812 - KIT33812ECUEVME. What needs to be done:</p> <ol style="list-style-type: none"> 1. Create a Load Box for testing the Evaluation Board 2. Create the control software to run the engine 3. Generate the Crankshaft and run the control software 4. Adjust parameters for gasoline, air and advance in order to run it on a real 1 cylinder engine 5. Run the Evaluation Board with 1 cylinder Engine 	<p>Electronics, Automatics A-level student Self-motivated, with strong learning abilities Basics understanding of engine systems Advanced knowledge of electronics, control systems, software algorithms, C programming language.</p>	C, HW	8
21	Electronic Engine Contr	<p>The student should run the Electronic Engine Control , monitoring parameters and measuring signals. The measurement results are used in the design of the new Electronic Engine sample.</p>	<p>Knowledge of electronic design, Power Electronics, Control systems / automatics.</p>	Hardware	4, 6, 8

Nr.c rt.	Title	Description	Requirements	Tests	hours / day
22	Manual Testing for OIP	<p>The IIC Business Unit provides software solutions for Multimedia Systems. The software solutions consist of SW packages integrated together in the system. The programming language used for writing these SW packages is C & C++.</p> <p>The student shall execute Manual Tests on different applications in the project, based on available test cases created beforehand. Possible applications in the area of Manual Testing are Connectivity, Bluetooth, Audio and Media, depending on the project needs.</p> <p>The student will have the support offered by a mentor during the project.</p>	SW Programming (any language - basic knowledge), logical thinking	one of ANSI C, C++, C# or Java	6
23	Automatic Generation of Test Environment for MCRP Aggregate	<p>Automatic generation of test environment required for testing the MCRP functionality on the test bench/at project integration phase. The developer will be introduced to our CessarCT environment and MCRP functionality.</p> <p>The practice consists in developing a number of jet files to generate the test environment required during tests on test bench and/or at integration into projects (e.g. source *.c files containing test interfaces and variables/calibrations, DDS *.grl files to be used in order to generate a test A2L etc.).</p> <p>The candidate required knowledge shall be: basic C programming skills and Java programming skills (medium knowledge would be a plus).</p>	C, Java programming language basic knowledge (medium knowledge would be a plus).	ANSI C, Java.	6, 8

Nr.crt.	Title	Description	Requirements	Tests	hours / day
24	SW Development for In	<p>Become familiar with the SW development for one of the most complex aggregate in P ES - Injection Realization. This is the control algorithm for Diesel piezo injector. In this project you will learn the development process, technologies and tools used in engine projects. All the process steps, technologies and tools will be used on real tasks. Scope of the practice is to understand the incremental development, the change and the configuration management and see the needs that are coming from various roles (SW Developer, Project Manager and Discipline Manager).</p>	<p>C languages Microcontrollers</p>	ANSI C	4, 6, 8
25	Stress test application for	<p>The proposed project needs to develop a solution to automatically test a multimedia system.</p> <p>The system includes an external radio that communicates with a 7inch display unit with a resistive touch via a serial connection.</p> <p>The project takes the form of a C++ application that can create, store and run various stress tests on the display unit. The application will use the API available on the display to simulate various user scenarios by sending a series of commands to the radio. The results will be collected and compared to another set of expected results.</p> <p>This test will help the development team by significantly shortening the time to validate a new software build. It will also improve the overall quality of the product.</p>	<p>- C++/OOP knowledge - C knowledge - Qt is a plus</p>	ANSI C, C++	4, 6, 8

Nr.c rt.	Title	Description	Requirements	Tests	hours / day
26	Application on an embed	<p>Implement an application using the internal API. Learn about working in a professional environment within Continental and gather knowledge and experience for your career.</p> <p>Using the API (and not only) you will:</p> <ul style="list-style-type: none"> - send/receive SMS - communicate on CAN - implement a graphical interface via QT - send/receive wireless packages - manipulate LEDs - play sounds - get the location from GPS <p>Everything is documented, so all you have to do is use the functions provided by the API.</p> <p>If you are eager to learn and have C++ knowledge, don't hesitate to apply.</p> <p>You will be guided the entire process and you will be following the Continental process as on a real project.</p>	C++, Qt would be a plus	microcontrollers, ANSI C, C++	8
27	Mobile Interfaces for Web	<p>This project has the goal to make a thorough analysis of how the new mobile technologies can fit into the existing web applications and to develop prototypes to help in deciding what the next steps in mobile area should be.</p> <p>Activities:</p> <ul style="list-style-type: none"> • Test the current applications behavior on mobile devices (Android / iOS devices) • Detect the UI problems and make improvement proposals • Develop a web mobile app for one Conti web app (e.g. TMT or eSign) 	OOP, C#/Java; knowledge of web and mobile development technologies (ASP.NET, MVC, Android, HTML5) is a plus	C#, Java	8

Nr. crt.	Title	Description	Requirements	Tests	hours / day
28	Intelligent TestBox build	<p>For the BCM' (Body Control Module) there was the need to build a Testbox which can handle a lot of signals. The goal was to build up a small box as possible with all needed functions and in many cases flexible for the use in other projects or for future function updates. There are in principle two signal directions: out of DUT into the Testbox (BDC-outputs for lights, motors,...) and the second one: out of Testbox into the DUT (BDC-inputs for switches, sensors,...) The DUT-outputs have a path through the Testbox and are scanned every 2ms. These infos are given to the LEDs in the front panels. All switches, potentiometers, hall-sensors, voltage sensors are simulated in the Testbox. There are three pcbs with simulations, beginning with simple relais simulating a normal switch, up to an electronic which simulates a potentiometers.</p> <p>The status of all outputs and simulations can be read out via the USB-interface. Also the Testbox internal simulations and buttons can be controlled via the USB-interface. For transfer settings between testboxes and for datalogger functionality there is a SD-card socket implemented</p>	HW related knowledge, soldering components, wire assembly	hardware and mechanics.	8
29	Design parts of an Instr	<p>Design in ProE the following parts as 3D model +2D drawing:</p> <ul style="list-style-type: none"> • Small light-guide - display • Main light-guide <p>If the ProE knowledge are not available, 1 week is needed for the basic training ProE.</p>	general technical knowledge / ProE	mechanics	6

Nr.c rt.	Title	Description	Requirements	Tests	hours / day
30	Design parts of a HUD	Design in ProE the following parts as 3D model +2D drawing: <ul style="list-style-type: none"> • Motor-holder • Bearing • Screen Top cover (acceptable only 3D model)	general technical knowledge / ProE	mechanics	6
31	General design + connectors	Design in ProE the following parts: <ul style="list-style-type: none"> • Actuator - 3D model + 2D drawing • Connector - 3D model + 2D drawing If the ProE knowledge are not available, 1 week is needed for the basic training ProE.	general technical knowledge / ProE	mechanics	6
32	Design parts/assembly	Design in Catia the following parts as 3D model +2D drawing: <ul style="list-style-type: none"> • Frame • Rear cover • Assembly If the Catia knowledge are not available, 1 week is needed for the basic training Catia.	general technical knowledge / Catia	mechanics	6
33	Event Data Recording test case generator with visual interface	Event Data Recording is a function in a vehicle that records the vehicle's dynamic, time-series data during the time period just prior to a crash event (e.g., vehicle speed vs. time) or during a crash event (e.g., delta-V vs. time), intended for retrieval after the crash event. We want to develop a test case generator for this functionality using Microsoft Visual Studio.	C programming language knowledge C#, C++	ANSI C, C#	4, 6, 8

Nr.crt.	Title	Description	Requirements	Tests	hours / day
34	Test Application for OVIP Media Wave2	The goal of this Project is to develop an Application which allows testing and validating the Business Logic code directly on the device before having the GUI available. It consists in developing a simple GUI, without respecting the customer specifications for that, but which allows testing completely the features like Gracernote which are introduced in Wave 2. The Project will be developed in Qt and will run on the OVIP device, under Linux.	C++ and Object Oriented Programming, - UML, Linux, basic Hardware and Media knowledge are a plus.	C++	6, 8
35	ATP infrastructure for SW Verification Tests within OVIP Media	The goal of this Project is to develop an infrastructure which allows the Business Logic code to be tested using automated tests (ATP). It consists in a layer which allows ATP to connect to the Media BL and access the interfaces which are available for other layers to be tested by.	- C++ and Object Oriented Programming, - UML, Linux, basic Hardware and Media knowledge are a plus.	C++	6, 8

Nr. crt.	Title	Description	Requirements	Tests	hours / day
36	Unitary Testing for OBD	<p>Into the Powertrain projects, before a project goes in production, a series of tests are done to ensure there are no interaction problems between the different functionalities included in the project. The ERRor Management (ERRM) functionality provides to all the other engine functionalities the interface to detect any error that can appear when the engine runs. It also provides the statistics used to determine the source of an engine failure.</p> <p>The Garage End of Line (GEOL) offers the possibility of different tests on engine components (injectors, actuators...) at the garage or the end of the production line.</p> <p>In this project you will learn the development process, technologies and tools used in engine projects. You will do unitary tests for ERRM and GEOL using RTRT language. This will allow you to build up testing competencies and to see complex C modules.</p>	ANSI C, Simulink (would be an advantage), algorithms, database	ANSI C	6

Nr.crt.	Title	Description	Requirements	Tests	hours / day
37	Hubble tool	<p>Today we are using many internal tools for Change and Configuration Management (e.g. IMES, LIMAS, IMS, ESI). It is difficult to have all the information in the same window or screen, and without to make a detailed search.</p> <p>Main purpose of this tool is to find and display all the links between Realization Order, Specification, Projects and Releases. All the links have to be displayed in a friendly manner.</p> <p>Second purpose of this tool is to find all problems (e.g. SAR, SPAR) link to a selected specification.</p> <p>The tool should be able to:</p> <ul style="list-style-type: none"> - connect to Limas database - connect to IMES database - connect to IMS database - create/maintain a defects database - connect to defects database - display and save the results 	ANSI C and Java (eventually C#), algorithms, database	ANSI C, Java	6
38	Basic software platform for microcontroller applications	The project consists of developing a basic software (BSW) platform for an ARM Cortex-MO+ microcontroller. The BSW platform offers components which are integrated in future projects, allowing the application access to libraries of functions which enable them to control the microcontroller's peripherals. Applications may vary from communication (SPI, I2C, Bluetooth drivers) to signal generation (motor control drivers), memory control (FLASH, DMA drivers) or even RTOS enhancements.	Good knowledge regarding microcontroller topics, good C programming skills. Digital logic understanding	ANSI C, microcontrollers	6, 8
39	Replacement of PMODE with XCP	Investigate opportunity of replacement PMODE module with XCP protocol (integrate XCP in project, and compare performance vs. existing PMODE in terms of runtime, ROM, RAM)	C knowledge, microcontrollers	ANSI C, microcontrollers	8

Nr.c rt.	Title	Description	Requirements	Tests	hours / day
40	ATP infrastructure for SW Verification Tests within OVIP Radio	The goal of this Project is to develop an infrastructure which allows the Business Logic code to be tested using automated tests (ATP). It consists in a layer which allows ATP to connect to the Radio BL and access the interfaces which are available for other layers to be tested by.	C++ and Object Oriented Programming, UML, Linux, basic Hardware	C++	6, 8
41	Development of Tests for the Automation Test Platform in OVIP Project	<p>The IIC Business Unit provides software solutions for Multimedia Systems. The software solutions consist of SW packages integrated together in the system. The programming language used for writing these SW packages is C & C++.</p> <p>Automation Test Platform (ATP) is a tool used to simulate the Human Machine Interface behavior in order to ensure coverage of all test suites. ATP tests are very important to ensure testing even for those features that do not have a complete implementation yet.</p> <p>For this purpose, the student has the develop software plug-ins, created on top of the interfaces offered by one software domain to a different software domain. Those plug-ins are used by the ATP Tool to create test suites and in this manner to test certain features and use cases.</p> <p>The student will have the support offered by a mentor during the project.</p>	Object-Oriented Programming (C++): basic level	C++	4, 6

Nr. crt.	Title	Description	Requirements	Tests	hours / day
42	Unit testing in OVIP project	<p>The IIC Business Unit provides software solutions for Multimedia Systems. The software solutions consist of SW packages integrated together in the system. The programming language used for writing these SW packages is C & C++.</p> <p>Unit testing is an important part of the software development process. In object-oriented programming, a unit is often an entire interface, such as a class, but could be an individual method, too.</p> <p>Unit tests are short code fragments created in C++ by programmers or occasionally by white box testers during the development process. Ideally, each test case is independent from the other test cases. Substitutes such as method stubs, mock objects, fakes, and test harnesses normally are used to assist testing a module in isolation. Unit tests are typically written and run by software developers to ensure that code meets its design and behaves as intended.</p> <p>Target of the activity is that a student creates a defined amount of unit test cases based on a design description. After the test case implementation, unit tests shall be run in order to check the compliance of the written code with the design. At the end, a test report must be</p>	Object-Oriented Programming (C++): basic level	C++	4, 6, 8

Nr. crt.	Title	Description	Requirements	Tests	hours / day
43	CAN & FlexRay testcase definition	<p>Tools: specific textboxes, bus monitoring tool.</p> <p>Daily tasks: Requirements understanding, test cases definition, test cases verification, reports.</p> <p>Testcase definition for CAN, FlexRay communication standard</p> <p>CAPL programming, CANoe usage</p> <p>A mentor will support you to stepwise take over own responsibility after an initial training phase which will familiarize you with our products, tools, processes and organization.</p>	<p>Basic C programming language knowledge</p> <p>Basic knowledge of embedded systems, knowledge about microcontrollers</p>	ANSI C, microcontrollers	4, 6, 8
44	Linux Development for ECUs	<p>Main tasks:</p> <ul style="list-style-type: none"> - porting existing configuration from kernel to a newer version. - building root file system with yocto. - setup system for HMI development. 	<p>Basic knowledge about Linux systems. C language is mandatory. C++ , python and Shell scripting is a plus.</p>	ANSI C	4, 6, 8
45	Data Logger Software	<p>Based on a hardware build around stm32f4 we would like to create a data logger that will analyze data in real time. Additional features could be added during the project.</p> <p>The project has two software packages:</p> <ul style="list-style-type: none"> - embedded development on stm32f4 - user friendly interface developed on PC 	<p>C language or C++ is mandatory</p>	ANSI C, C++	4, 6, 8
46	Checking of data declaration tool	<p>Tool will check if specified data with their defined ranges, are defined correctly within their corresponding GRL file. This tool will be useful during Code Object Review phase. Now this checking is done manually, once with the implementation of the tool, checking will be done automatically, which will result in avoiding of human errors during manual verification.</p>	<p>Visual C++</p>	C++	6

Nr.c rt.	Title	Description	Requirements	Tests	hours / day
47	3G On Board Unit Communication Module	<p>The main target of the project is to create a software module inside an existing project which enables 3G communication features to the project.</p> <p>The starting point will be the existing 2G communication module which uses a Cinterion AGS2 modem. After this starting point the software module must be modified to communicate with an AHS3 modem which will enable 3G communication.</p> <p>During the project the participant will earn experience in the following subjects: GSM Modems, Embedded Systems, Web data transfer protocols (JSON), TCP, Internet Of Things, GPS, SSL/TLS.</p>	Serial Communication, Microcontrollers, ANSI-C, Basic TCP communication	ANSI C, microcontrollers	8
48	Signal interpreter plugin	<p>Background: Logic analyzers are used in the daily work in the Low Level Embedded area for analyzing different communication protocols (I2C, CAN, UART, etc). The logic analyzer used in the I CV&AM Low Level team comes with a standard GUI that has a set of predefined signal interpreters included.</p> <p>Project proposal: As an extension to the provided signal interpreters, a library containing signal processing methods can be done as plugins. Specific Continental protocol (InterControllerCommunication) and other signal processors (e.g. PWM) need to be implemented in order to be used in the logic analyzer GUI.</p>	C++, C#, XML language, basic embedded knowledge (signals, communication protocols)	ANSI C, microcontrollers, C++, C#	4,6

Nr.c rt.	Title	Description	Requirements	Tests	hours / day
49	Dynamic Regression Test implementation	Development of a dynamic test script in python to cover a given requirement. The script will have to use the API's of TAS test equipment to fulfill the test cases preconditions, input values and perform the evaluation using python function to determine automatically the status of the test cases.	python programming or other OOP	C++, microcontrollers	6
50	Display test tool chain	Background: The Commercial Vehicles Instrument Clusters usually have a display for providing information interface to the driver. Several display types are supported: monochrome dot matrix or TFT color displays. Project proposal: A tool chain must be created that allows automatic regression testing of graphical subsystem. This would need to have an interface to the graphical design tool where all the graphic objects are created/integrated (PC) and also to the embedded target used for displaying the objects. Objects generation, data acquisition, storing and comparing of outputs should be provided by the tool chain. In the end the effort spent for regression test of the graphical subsystem should be reduced.	C, C++, C#, XML language, basic embedded knowledge, CAN	ANSI C, microcontrollers, C++, C#	4, 6

Nr.c rt.	Title	Description	Requirements	Tests	hours / day
51	Final Test Report automatic generation	<p>Tool for automatic creation of a Final Test Report document.</p> <p>The application shall use a word document called DTR - detailed test report - to generate another file called FTR - Final Test Report.</p> <p>The FTR has a fix structure and will use information from initially selected DTR ad also from a database.</p>	<p>SQL / .NET / XML / Algorithms / Logic / Graphical User Interfaces / Threading / Microsoft Office</p> <p>Ability to read, understand and follow a basic set of requirements</p>	C++, C#	6, 8
52	Transmission Control Unit (TCU) Software Real-time analyzer	<p>Real-time analysis of a TCU software program including: Program Trace Performance Analysis Code Coverage using analyzer capabilities of a tool like iSYSTEM winIDEA or On-chip Debug Support feature of the microcontroller.</p> <p>Create a TCU Program Analysis Framework by creating templates for TCU specific analysis needs.</p> <p>The project offers the student the opportunity to analyze and understand many aspects of an embedded software program.</p>	<p>C language Microcontrollers</p>	ANSI C, microcontrollers	4

Nr.c rt.	Title	Description	Requirements	Tests	hours / day
53	Automation of Instrument Cluster Functional Tests	<p>Develop a test environment (C#), integrating test cases for the complete set of features supported by an instrument cluster (stepper motors, display, communication interfaces, digital inputs and outputs, etc), that will command requests and evaluate incoming responses (CAN interface based) from instrument cluster in order to automate as much as possible the functional testing process.</p> <p>Main tasks:</p> <ul style="list-style-type: none"> - Understand the instrument cluster requirements / features set and update logiCAD test application (model based development) - Develop a PC graphical interface and implement test cases for the functional tests - Parse input files (.xml) to automate tests cases selection for the instrument cluster supported features; generate output files (test reports) that should be linked to the functional test cases written in DOORS 	C, Embedded systems / Microcontrollers, Basic hardware knowledge & handling of electronic equipment (e.g. oscilloscope, multimeter)	ANSI C, microcontrollers, C#	8

Nr.c rt.	Title	Description	Requirements	Tests	hours / day
54	Collecting data from external applications into a db	<p>Reports need to be created in a specialized tool using data from different applications and files.</p> <p>The project is intended to make an analysis of all data that needs to be included in the reports and how it can be automatically moved into a database.</p> <p>The outcome of the project will be the implementation of the database that will contain the data and the synchronization between the database and the external sources.</p> <p>The project outcome shall consists of:</p> <ol style="list-style-type: none"> 1. A description of all sources for data to be collected 2. A database structure for collected data 3. Solutions for collecting data from each source defined in point 1 4. Implementation of 2 and 3. <p>Further, the database will be used to generate monthly reports but this activity is not in the scope of the current project.</p>	<p>Databases / SQL / XML / Algorithms / Logic / Graphical User Interfaces / Threading / Microsoft Office</p> <p>Ability to read, understand and follow a basic set of requirements.</p>	C++ or C#	8
55	Simulink model improvement with Matlab script	<p>The main target of the project is to create Matlab scripts that will improve the Simulink models designed for Automotive Application: adding block, organizing signals, altering existing implementation.</p>	Ansi-C, Matlab, Simulink	Matlab-Simulink	8

Nr.c rt.	Title	Description	Requirements	Tests	hours / day
56	CAN and LIN Gateway	<ul style="list-style-type: none"> - the purpose of the project is to design and build a gateway box with multiple CAN and LIN interfaces to be used for driving more devices from one common control unit - hardware students will have to develop the block diagram, schematic and layout of the PCB, and finally assemble the components; - mechanical students should design and build / 3D print an enclosure to host the PCB and also design the harness cables to all the devices - all students will get knowledge in the field of automotive engineering, communication interfaces and specific challenges of the automotive environment 	<ul style="list-style-type: none"> - Hardware: basic knowledge of analog and digital electronics, ECAD environment - Mechanics: basic knowledge of mechanical design, MCAD (ProE or Catia V5) 	Hardware and mechanics	8
57	Dynamic graph visualization of the base software architecture in an instrument cluster project	<p>Develop a PC software tool (C#) which generates a graphical overview of the embedded software architecture for an instrument cluster base software project. The tool will use as input configuration files (pseudo language format) of the base software modules.</p> <p>Main tasks:</p> <ul style="list-style-type: none"> - Understand software architecture of an instrument cluster base software project - Parse input files and store useful information in internal data structures - Develop a PC tool which is using existing library for graphs representation to visualize in a dynamic way details about software architecture 	ANSI C, Embedded systems / Microcontrollers	ANSI C, Microcontrollers, C#	8

Nr. crt.	Title	Description	Requirements	Tests	hours / day
58	SW system test for the display cluster (display, pointers, tell tales)	<p>The existing test specifications are created according to the customer requirements. They reflect different use cases in which the correct behavior is checked.</p> <p>Having a test bench made out of SW tools and HW equipments a real car is simulated. The tester will have to modify the simulation and check that the result is as expected.</p> <p>For example if you simulate that the tank lid is open, the expected result is that a warning for the driver appears.</p> <p>The tests are performed for the following customers:</p> <ul style="list-style-type: none"> - Mercedes - VW - Ford - Fiat 	<p>Electronics</p> <p>Basic programming knowledge</p> <p>Good Microsoft Office knowledge</p>	<p>ANSI C, microcontrollers, Hardware</p>	6,8
59	Tooling enhancements	<p>Autosar standard is used in all Electronic Control Units distributed in the car. The software is split in 2 parts: Applicative and Basic. The basic part is developed according with AUTOSAR standard. The integration of the Basic software is facilitated by applications developed based on Eclipse framework.</p> <p>The theme is to enhance the development process by creating add-ons for the various tools that we use in the development process.</p> <p>You will implement small Java applications (standalone and/or web based).</p> <p>For your daily work you have to work with professional SW development tools based on Eclipse.</p>	<p>Java, XML, Maven, Eclipse, Tomcat, Apache Velocity</p>	<p>Java</p>	4,6

Nr.c rt.	Title	Description	Requirements	Tests	hours / day
60	Eclipse based application for integration of AUTOSAR SW	<p>Autosar standard is used in all Electronic Control Units distributed in the car. The software is split in 2 parts: Applicative and Basic. The basic part is developed according with AUTOSAR standard. The integration of the Basic software is facilitated by applications developed based on Eclipse framework.</p> <p>The theme is to develop an application based on Eclipse used for the integration of AUTOSAR SW in the real customer project.</p> <p>You will implement an application in Java language through all SW development steps: requirements, design, coding and testing.</p> <p>For your daily work you have to work with professional SW development tools based on Eclipse</p>	Java, XML, ANT, Eclipse	Java	4, 6
61	Tools automatic installation	The project scope is to develop a program which is able to search for the tools and algorithm new releases and to start the installation on user request. The program should run in background on user's computer.	Scripting language (Perl)	ANSI C, C++, C#	4, 6, 8
62	BSW support application	Come and create an application that will help our team to support our clients better. The application will allow a client to enter and track his support request, as allows our team to view, answer and solve the requests. The application shall have a simple interface with complex functionalities behind.	Java, Databases	Java/C++	4, 6

Nr.c rt.	Title	Description	Requirements	Tests	hours / day
63	Fee Viewer	Flash memory dump analyzer: 1.The tool will read from a dump file and it will identify block headers, bank headers and block data and it will output information in a html format; 2. Convert the dump file into a .c file that can be used for reproducing the flash banks in RTR testing tool.	Java, Pearl, Html	ANSI C, Java	4, 6, 8
64	High precision measurement device design	The subject of this thesis consist in designing of a device which will be used in Powertrain Sensors and Actuators Timisoara for high precision positioning for microscope observation and measurements in daily tasks.	CAD, machining, technical drawing	mechanics	8
65	Adaptive lights system	The student will have to design and develop a system which reads the ambient light intensity and the distance to the car in front, and dependant of the read values will have to adapt the intensity and the position of a car lights A photo resistor will be used for reading the ambient light intensity and a ultrasonic distance sensor will be used to read the distance to the car in front. The lights system will be a LED system because it is easier to control the intensity of the light. The position of this lights system will be mobile and it can be vertically adjusted using an electrical motor.	ANSI C, Microcontrollers, Electronics	ANSI C and microcontrollers	6, 8

Nr.c rt.	Title	Description	Requirements	Tests	hours / day
66	Reprogramming protocol for Cluster Instruments	<p>The scope of the project is to implement the Application Layer of a protocol able to flash memory devices present on a cluster instrument. The supported memory devices are:</p> <ul style="list-style-type: none"> - internal flash memory of microcontroller - internal RAM memory of the microcontroller - external NOR flash memory on serial bus SPI <p>The access to memory devices should be based on usage the existing software drivers for every memory device.</p>	C , microcontrollers, basic electronics	ANSI C, microcontrollers	6, 8
67	Instrument cluster demo for vehicle test chassis	<p>Design and implementation of a several SW packages that handles the acquisition and processing of vehicle functions. Based on the data gathered from the vehicle functions/sensors via CAN, the instrument cluster should be able to display information on a TFT display.</p> <p>Main tasks:</p> <ul style="list-style-type: none"> - Understand of vehicle / instrument cluster architectures; - Work with specific automotive tools; - Design and implementation of the application module that handles the acquisition of data from the car; - Design and implementation of the HMI (Human-Machine-Interface) module that handles the graphics part of the instrument cluster; - Integration and testing of the newly created modules in the test chassis vehicle; 	C / C++, microcontrollers, basic electronics, operating systems	ANSI C, microcontrollers, C++, hardware and mechanics	6, 8

Nr. crt.	Title	Description	Requirements	Tests	hours / day
68	Driver information environment	<p>Development of a simulated driver environment using the following components:</p> <ul style="list-style-type: none"> • Driver seat (with digital control) & non-reflective windshield • Audi color cluster instrument • Audi Head-Up Display • Driving wheel with digital controls • Driver pedals with digital control <p>Activities</p> <ul style="list-style-type: none"> • Development of a SW package for the cluster instrument, head-up display and the board controlling the driving wheel and pedals so that the driver is able to change parameters (e.g. acceleration) which will be afterwards displayed on the cluster instrument & head-up display accordingly. • Code debugging on each unit <p>Test of each unit and the whole system.</p>	C / C++, microcontrollers, basic electronics, operating systems	ANSI C, microcontrollers	6, 8

Nr.c rt.	Title	Description	Requirements	Tests	hours / day
69	OpenVG Mini-Framework for Instrument Clusters with 2D Animations	<p>Instrument clusters and secondary displays are becoming more and more advanced with more and more information to present in an eye-candy way to the driver.</p> <p>In this project, the target is to develop a basic framework based on OpenVG to allow easy development for reconfigurable instrument clusters (full TFT clusters). At the end of the period, the following features should be provided by the framework:</p> <ul style="list-style-type: none"> - Round 2D animated pointer gauges creation with specified size, position and texture - Arc 2D animated pointer gauges creation with specified size, position and texture - Bitmaps display with specified size, position and transparency - Text display with specified size, position and transparency - Provide 2D animation support for displayed objects 	OpenGL (beginner), C/C++, Linux (beginner), Raspberry PI	ANSI C, C++	6

Nr.c rt.	Title	Description	Requirements	Tests	hours / day
70	OpenGL Mini-Framework for Instrument Clusters with 3D Animations	<p>Instrument clusters and secondary displays are becoming more and more advanced with more and more information to present in an eye-candy way to the driver.</p> <p>In this project, the target is to develop a basic framework based on OpenGL to allow easy development for reconfigurable instrument clusters (full TFT clusters). At the end of the period, the following features should be provided by the framework:</p> <ul style="list-style-type: none"> - Round 3D animated pointer gauges creation with specified size, position and texture - Arc 3D animated pointer gauges creation with specified size, position and texture - Bitmaps display with specified size, position and transparency - Text display with specified size, position and transparency - Provide 3D animation support for displayed objects 	OpenGL (intermediate), C/C++, Linux (beginner), Raspberry PI	ANSI C, C++	6
71	Aggregation of Power Take-off functionality	<p>Get a basic understanding of function development process and tools.</p> <p>Obtain an overview of the Power Take-Off functionality and restructure it into smaller, separate components, in order to have a more consistent and modular base for future development.</p>	Matlab/Simulink	Matlab/Simulink	4

Nr. crt.	Title	Description	Requirements	Tests	hours / day
72	Migrate Warping simulation GUI from GLUT to QT	<p>A PC simulation used to validate the warping algorithms in the HUD projects currently has the GUI (Graphical User Interface) built on GLUT (an OpenGL based library). As this approach is deprecated in Windows based development environments there is a need to migrate the GUI to a more actual, still cross-platform technology. The proposal is to use QT for this.</p> <p>Activities</p> <ul style="list-style-type: none"> • C++ development with QT in Windows environment • Code debugging 	C++ (mandatory), QT (experience with the QT technology is a plus)	C++	6, 8
73	Rework of display objects handling in Roadmap projects (target module DOF)	<p>The DOF module (Display Output Functions) used in many projects is built on the multi-branching coding technique (i.e. switch-case, if-else). This reduces the code maintainability. A better approach is to use look-up tables that would contain also the different conditions needed to activate specific outputs. The final result would reduce the source code length a lot. The implementation must be validated with the module specific unit test.</p> <p>Activities</p> <ul style="list-style-type: none"> • C development • Code debugging • Test on target 	C, operating systems	ANSI C, microcontrollers	4, 6, 8

Nr.c rt.	Title	Description	Requirements	Tests	hours / day
74	Test Tool Generator	<p>The project aims to create a Web-based solution generating automatically tests for candidates who apply for open positions or summer practice jobs. The solution allows users to create and store questions, as basic elements, for testing candidates' knowledge in different areas. The data area stored in a dedicated database. The application main goal is to generate one or more tests containing questions from the backend database, which would fulfill certain input criteria:</p> <ul style="list-style-type: none"> - Number and name of sections in the test - Selected difficulty/complexity of included questions - Amount of time necessary for completing the test <p>The application will have several other features: storing generated tests, possibility to customize tests, database and application security handling, test profile storing etc.</p> <p>The application shall be developed in C#/ .NET and uses a MS SQL database.</p>	<p>OOP, OOD, at least one object-oriented language (C# is preferred). Relational database knowledge.</p>	one of C#, Java or C++	4, 6
75	SQA optimizations	<p>Development of tools in a OOP language to optimize SQA tasks.</p> <p>The project will generate automatically several reports for the Quality Engineer, based on information collected from documents stored in Configuration Management systems.</p>	<p>OOP - at least one object-oriented language, OOD.</p>	one of Java , C# or C++	4, 6

Nr.c rt.	Title	Description	Requirements	Tests	hours / day
76	Diagnosis Tester Emulation on an instrument cluster	<p>Development of a diagnostic virtual tool on a cluster instrument that will act like an external offboard diagnostic tool and will interrogate the cluster instrument itself or any other ECU inside the car using UDS messages. The control of the diagnostic command and parameters will be done using a menu.</p> <p>Activities:</p> <ul style="list-style-type: none"> • Development of a SW package for the cluster instrument, so that the driver is able to perform self diagnostic • Code debugging • Testing the entire system using different testing techniques 	C / C++, microcontrollers, basic electronics, operating systems	ANSI C, microcontrollers	6, 8
77	Integration of Continental change management system with OEM systems	<p>Project will synchronize data between Continental change management systems with OEM systems based on Service Oriented Architecture (SOA). During project students will learn how to build web services and integrate business workflows.</p>	Java/.NET Advantage: Xml Webservice	C#, Java	4, 6
78	Software Updates Management Application	<p>Develop an application that manages the software updates available for a specific device and based on the user's input it does the required update.</p>	Object Oriented patterns, relational database and experience with Client - Server application development	C++, C#	6
79	Innovative telematics system	<p>Android based application used to remotely control telematics devices installed in vehicles Inspired user interface Optimized short range Bluetooth/WiFi communication protocol On board telematic data recorder device (Embedded C++ application)</p>	Android application development C++ OOP	C++, Java	4

Nr.c rt.	Title	Description	Requirements	Tests	hours / day
80	Road Signs recognition micro vehicle	Building a micro vehicle to be able to recognizes Road Signs. Micro vehicle will be able to: - Control at least two wheels by DC motors using ARM control platform. - To use a smart phone or build in video camera module with serial /parallel communication for signs recognition. - Follow the road rules depending of the sign seen. - Recognize at least four road signs	Android Java, ANSI C, Microcontrollers, Electronics	ANSI C, microcontrollers, Java	6, 8
81	Car window sun blocker	The device has a sensor unit for determining the vehicle environment, an evaluation unit for processing data of the sensor unit, a component for preventing the glaring and a control unit for controlling the component. The evaluation unit determines a situation of glaring by the sun on the basis of the data of the sensor unit. The control unit controls the component depending on the situation of glaring determined by the evaluation unit	ANSI C, Microcontrollers	ANSI C, microcontrollers	6, 8
82	Mechanical Design and Documentation Support for Airbag Control Unit	Documentation update (FMEA, requirements management, update overviews) Design tasks of low complexity (housing design considering guidelines) Tolerance calculation Supplier tracking Support for technical drawing preparation	- ProEngineer/Creo	mechanics	4, 6
83	SW Architecture for Euro NCAP ADAS Camera	Development of SW architecture for Euro NCAP ADAS camera. Selection of relevant requirements, creation of SW architecture based on system configuration, definition of SW/Algo interfaces.	Image processing knowledge (basic), ANSI C/C++/ C#, microcontrollers, Office, Visio (Flowcharts/Diagrams)	microcontrollers, C++, C#	6, 8

Nr.c rt.	Title	Description	Requirements	Tests	hours / day
84	HW Concept for ADAS Euro NCAP Camera	Create a HW module concept for a camera project. Analyze existing requirements, reverse engineering and benchmark, Mathcad calculations, PSPICE simulations and power dissipation. Test bench measurements for a specific module		microcontrollers, Matlab, Labview, PSPICE, hardware basics	6, 8
85	HW device for transferring Images from ADAS Cameras to a remote Electronic Unit over a high speed automotive wired channel. (USB or LVDS)	<p>The ADAS HW team is developing Camera sensors for ADAS functions and for the future Autonomous Driving vehicle. One needed functionality is to transfer the images from existing ADAS Camera sensors to a remote Processing unit that will be further used for high computation power algorithms. The actual ADAS Camera sensors provides an HW interface where external devices can be connected in order to access the Image sensor Data.</p> <p>The device must be connected to the interface, to convert the image data in the corresponding format (USB or LVDS) and transfer the image data to the other end.</p> <p>Technical Mentoring by ADAS HW experts will be ensured for this project. The students will have the possibility to learn about the electronic designs of Camera sensors and the functionality of the CMOS Image Sensors.</p>	Digital and Analog Electronic	HW + Microcontrollers	8

Nr. crt.	Title	Description	Requirements	Tests	hours / day
86	VC++ application for sending Images from ADAS Camera Sensors to a remote Computer via 3G/LTE modems	<p>The ADAS Advanced Engineering team is developing functions for the future Autonomous Driving vehicle.</p> <p>A needed functionality is to send images from ADAS Camera sensors to a remote computer. Actual the ADAS Camera sensor, in the test vehicle, is connected to a powerful computer used for recording data during test drives.</p> <p>The application that has to be developed is to store one image from the video stream when a send command is triggered, and then to send the stored image to a remote PC via a 3G/LTE modem.</p> <p>Technical Mentoring will be ensured by ADAS experts in the area of ADAS Advanced Engineering technologies.</p> <p>The students will have the possibility to learn about the ADAS technologies for the Automated Driving vehicles.</p>	VC++	C++	8
87	Restructure of ADAS Sharepoint page	<p>Current version of Sharepoint for ADAS BU needs a restructure of directory, content and view. The responsible will gain knowledge on Sharepoint via the trainings provided in the company and will have the chance to apply this new information in the re-organizing of the internal page used in the BU.</p>	basic programming, basic HTML and Web development knowledge	no test requested	4

Nr. crt.	Title	Description	Requirements	Tests	hours / day
88	SPICE simulation of ISO 16750-2 conducted electrical transients	<p>Electrical disturbances occur during normal operation of many items of motor vehicle equipment. They are generated over a wide frequency range, and can be distributed to on-board electronic devices and systems by conduction, coupling or radiation.</p> <p>The scope of this project is to simulate these disturbances in order to predict the behavior of the DUTs before subjecting them to testing, thus reducing the possibility of failures.</p>	P Spice or other SPICE simulation tool, Analog electronics	Hardware Test	4