Multidisciplinary Learning through Implementation of the DVB-S2 Standard

Yuri Murillo, Bertold Van den Bergh, Jona Beysens, Alexander Bertrand, Wim Dehaene, Panagiotis Patrinos, Tinne Tuytelaars, Ruth Vazquez Sabariego, Marian Verhelst, Patrick Wambacq, and Sofie Pollin

We give an overview of the objectives of the EAGLE project and the relevance of the DVB-S2 standard, the design and implementation task that the students need to accomplish, and how they finally learn to master the complexity of a real communication standard, VHDL coding, and system verification and integration.

ABSTRACT

Telecommunication standards are documents that contain consolidated knowledge about communication systems and implementation best practices. They are created based on long consensus processes in order to meet practical constraints. This article describes how the DVB-S2 standard is used in the electrical engineering curricula at KU Leuven within a design and implementation course called EAGLE. The goal of the course is to teach third-year Bachelor students how to apply the abstract knowledge gathered in the theoretical courses (control theory, communication theory, software, and hardware) to the design of a complex and real engineering project. Based on their progress during the academic year, we illustrate how standards can be used to teach a broad skill set and act as an accelerator to abstract system implementation complexity. We give an overview of the objectives of the EAGLE project and the relevance of the DVB-S2 standard, the design and implementation task that the students need to accomplish, and how they finally learn to master the complexity of a real communication standard, VHDL coding, and system verification and integration.

INTRODUCTION

The training of electrical engineering students should be broad and multidisciplinary, providing strong theoretical and practical foundations. However, it is challenging to design an engineering curriculum or even a single course that forces students to simultaneously think at the system level, develop theoretical insight on the underlying topics, and master practical implementation. As standards are documents culminating years of research efforts and must be written in coherent and unambiguous terms, they can potentially be a unique means of teaching students how theoretical knowledge is translated into real-world applications. Moreover, it can help them identify industry leaders and understand the business aspects that rule standardization before entering the market once their education is completed.

Contrary to this idea, engineering education at the Bachelor level classically omits the study of standards as they are often considered to be too specific for one particular technology or to contain too many practical details. As a result, courses cover only abstract knowledge and understanding of common technologies, and application is postponed to the Master level. During the first years of their education, students obtain a skewed view of the engineering field, making them unaware of the real competencies needed for engineering, which are covered at the Master level [1, 2].

In this article we explain how we used the second generation of the digital video broadcasting over satellite (DVB-S2) standard [3] in the design of the EAGLE course, taught to third-year Bachelor students. This standard is used to implement a wireless video link for a drone within a total time budget of around 50 hours using a lightweight and widely used field programmable gate array (FPGA) board, the ZyBo [4]. The main task is the partial implementation of a DVB-S2 transmitter in the very high-speed integrated circuit (VHSIC) Hardware Description Language (VHDL). Aware of the importance of introducing project-based learning at the Bachelor level [5], we decided to avoid designing a completely made up project and exploited the benefits of using an already existing standard: students are able to understand its importance faster and become more motivated thanks to its connection to the real world [6].

We now summarize the main arguments in favor of our choice of standard.

•As the task that the students need to fulfill is the development of a wireless video link, it makes sense to select one of the main technologies used nowadays. DVB-S2 is currently the main satellite digital television broadcast standard, used worldwide to broadcast standard definition television (SDTV) and high definition television (HDTV) services. The first part of [7] gives a clear overview of the advantages of DVB-S2 over legacy systems. More advanced standards could have been used (e.g., the terrestrial extension known as DVB-T2), but they were too complex for the technical background of third-year Bachelor students.

•It covers content that is already studied or will be studied for the degree.

•Off-the-shelf components can be used for testing the implemented code, such as the ZyBo FPGA board for the transmission and a standard-compliant DVB-S2 receiver for the reception of the video signal. Additionally, as these components are relatively cheap, students are not intimidated by them and feel free to experiment without the fear of breaking an expensive device.

•While DVB-S2 is a common and widely used standard, no open source, complete VHDL implementation is available online. This means that students get the opportunity to play a central role

Yuri Murillo, Berthold Van den Bergh, Jona Beysens, and Sofie Pollin are the primary authors of this article.

Digital Object Identifier: 10.1109/MCOM.2017.1700007

The authors are with KU Leuven

0163-6804/17/$25.00 © 2017 IEEE

Authorized licensed use limited to: KU Leuven Libraries. Downloaded on March 31,2021 at 08:08:01 UTC from IEEE Xplore. Restrictions apply.

in the implementation of the standard, while also learning a new programming language in the process.

The structure of this article is as follows. First, more details of the EAGLE course are given. Next, we give in-depth information about the wireless video link task. We discuss the conclusions we can draw after running the course for one year. Finally, we conclude this article, where we give examples of other accents that could be given with the same material. The whole project gives the instructor a wide palette of possible study items on which to elaborate.

Top-Level Overview of the EAGLE Course

In the third year of the electrical engineering Bachelor program at KU Leuven, there is an important design and implementation course aiming to teach students how to bring this broad theory into practice, called EAGLE. We introduce this project and give an overview of the different modules to be implemented by the students, while we describe the wireless video link task related to the DVB-S2 standard. Finally, we elaborate on the different resources available to the students for completing the course and its evaluation criteria.

The EAGLE Mission

The EAGLE problem solving and design course carries a total study load of 270 hours per student, representing nine European Credit Transfer System (ECTS) credits out of the 60 envisioned for the academic year [8]. In the current EAGLE course, nine teams of ten students have to design a smart drone able to fly autonomously and execute a complex task: the EAGLE mission. Note that a different drone per team should be designed. In order to do so, a total of six modules need to be implemented:

1. Hardware design of the electronic control system of the engines
2. Vision processing to recognize the environment
3. Cryptography to secure the mission
4. Autonomous navigation and control
5. Wireless power transfer
6. Wireless video link for transmitting the video stream obtained from a camera to a central controller

This last module represents 50 out of the 270 working hours dedicated to the project.

The EAGLE mission is demonstrated during the final demo session of the course. A safety net ensures physical separation between the drone and the students, and a 5 × 4 grid of quick response (QR) codes is placed on the floor, as well as a landing platform. This mission takes place in an indoor scenario and is divided into three different phases, as can be seen in Fig. 1. An increasing number of modules is needed for successfully completing each phase, with all of them correctly performing to ensure that the mission is finally accomplished.

In the first phase students need to manually fly the drone toward the first QR code coordinate using a remote controller and based on the camera signal sent through the wireless video link to a receiver PC. The steering of the motors and the navigation controller should also be operative in this phase in order to ensure a stable flight. In the second phase the drone switches to autopilot mode and flies over the grid describing a predefined path. This path is obtained from the QR codes, which contain encrypted information about the next coordinate to which the drone should fly. For this phase the cryptography, autopilot, and vision processing modules need to be functional in order to detect the lines of the grid and decode the QR information. Finally, in the third phase the drone obtains the position of a landing platform surrounded by a copper coil that is connected to a power receiving circuit, able to power a light emitting diode (LED) meter. The drone then needs to autonomously land on the platform and inductively transfer power from its coil to the one underneath the platform. Based on the performance of the landing maneuver and the wireless power transfer module, the LED meter will display a higher or lower level, and finally the mission is accomplished.

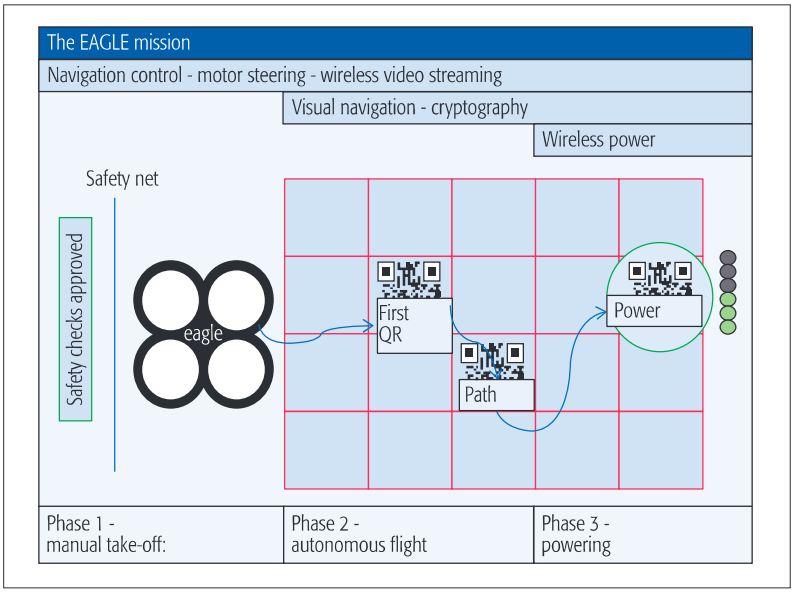

Figure 1. The EAGLE mission featuring the three described phases, where an increasing number of modules need to be operative to successfully perform each task.

The Wireless Video Link Module

The wireless video link module is the focus of this article, and covers about one-sixth of the course. During the first semester two to three students per team work on a partial VHDL implementation of a DVB-S2 standard-compliant transmitter. This code will be deployed on the ZyBo board installed in the drone. The video stream is transmitted to an off-the-shelf DVB-S2 receiver in order to obtain the signal being recorded by the camera.

The ZyBo development board was chosen for this project due to its low cost and small learning curve. This board has several I/O ports (USB, HDMI, Ethernet, VGA, audio, and GPIO), two ARM Cortex-A9 CPUs, and one FPGA. The camera functions like an IP surveillance capture device: H264 encoded packets are sent via Ethernet to the ZyBo at a rate of 1.5 Mb/s and encapsulated into a unidirectional lightweight

Authorized licensed use limited to: KU Leuven Libraries. Downloaded on March 31,2021 at 08:08:01 UTC from IEEE Xplore. Restrictions apply.

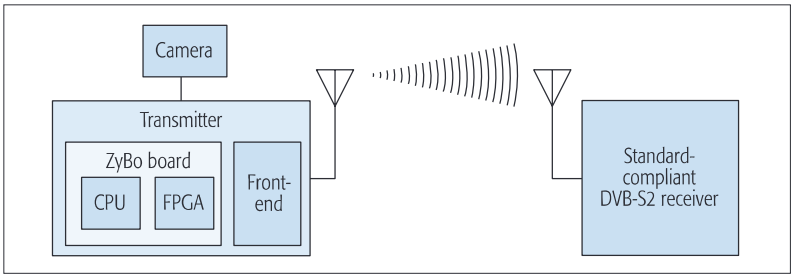

Figure 2. Architecture of the wireless video link, consisting of a ZyBo board with added camera and custom front-end for the drone and an off-the-shelf DVB-S2 receiver for the ground station. The FPGA on the ZyBo board runs the custom DVB-S2 implementation.

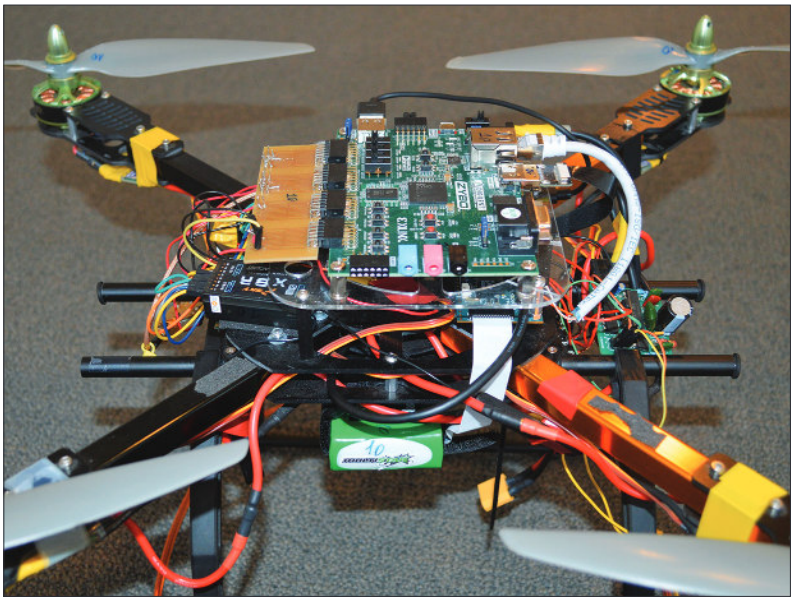

Figure 3. EAGLE drone with the ZyBo board on top of it. The camera is attached below, facing the ground in order to recognize the QR codes.

encapsulation (ULE) stream [9] which has an acceptable overhead [10]. This MPEG transport stream is then handed to the DVB-S2 transmitter core in the FPGA fabric. The IP core is able to process approximately one bit of the transport stream data per clock cycle. Since it is capable of running at up to 100 MHz, much higher throughputs are possible. The IP core supports all standard code modulations: quadrature phase shift keying (QPSK), 8-PSK, 16-amplitude and phase shift keying (APSK), and 32-APSK. All code rates are supported: 1/4 (short block only), 1/3, 2/5, 1/2, 3/5, 2/3, 3/4, 4/5, 5/6, 8/9, and 9/10. In the case of the EAGLE project, the output signal has a symbol rate of 866 ksamples/s. A root-raised cosine (RRC) filter with a roll-off factor of $\alpha = 0.35$ is used. This digital baseband stream is finally delivered to a custom designed front-end that transmits it in the 5.8 GHz industrial, scientific, and medical (ISM) band. For the EAGLE course, the constant coding and modulation (CCM) mode is used along with the regular single transport stream broadcasting service configuration. This architecture is depicted in Fig. 2.

Other low-cost boards for the Xilinx Zynq-7000 system on chip (SoC) can be used instead of the ZyBo, such as the ZedBoard [11]. The camera selected for this project is a regular Raspberry Pi camera [12], while the receiver chosen is the TBS5980 QBOX CI DVB-S2 TV tuner [[13], which is connected to a PC. An EAGLE drone showing this hardware is depicted in Fig. 3.

The wireless video link task is divided into two different phases: design and implementation. In the initial design phase, students need to become familiar with the DVB-S2 standard and understand the functionality of the transmitter sub-blocks. They receive a copy of the full DVB-S2 standard, as well as a description of a partial VHDL transmitter implementation. This is a version of a fully functional custom implementation of the DVB-S2 standard for the ZyBo FPGA,[1] with some blocks omitted that they will have to implement in the second phase. This approach reduces the complexity of the task and ensures that students are not overwhelmed by the standard and the complexity of VHDL coding, while still leaving space for a sense of achievement and experimenting with the results of their work. The software and drivers for controlling and configuring the transmitter are also given to them, and run on the CPU. The software can dynamically modify the code rate and modulation parameters of the DVB-S2 transmitter and calibrate the front-end, while the drivers manage the communication with the camera.

We emphasize that this approach is tailored to the technical background of third-year Bachelor students. They have already covered all mathematical background and been introduced to topics such as digital signal processing, programming, and circuit design. However, they are not familiar yet with error coding, modulation theory, propagation issues, or VHDL. If the EAGLE course was designed for the Master level, much less or no content would need to be provided.

Evaluation Criteria and Resources Available to Students

In the EAGLE course every group needs to autonomously elaborate a plan for the design of the six modules and justify the amount of resources (team members) dedicated to each one of them. No information is given to them in this regard, but as the wireless video link module needs to already be operative in the first phase of the EAGLE mission, it is expected that all groups prioritize this task from the beginning, dedicating two to three students for its implementation.

Regarding grading, every group is evaluated based on both technical (meeting a predefined set of deadlines) and non-technical aspects (managing skills, out-of-the-box thinking, work ethics), and several demo sessions are expected during the year. An online platform stores all material given to the students for the design of each module. Inside this platform every group also has a blog in which they have to explain their progress and fulfill a peer review assessment, from which the individual grade is obtained in combination with the group grade. Knowledge transfer among team members is done through their regular meetings and the online platform, where several question and answer (Q&A) sessions are additionally planned with the teaching staff after the weekly sessions.

There are six technical experts involved in the course, advising students with respect to each of the six technical modules. In addition, each team

[1] The open source implementation of the DVB-S2 transmitter developed for this project can be found at http://opendvb.org. Any code related to parts students need to implement is available only upon request.

Authorized licensed use limited to: KU Leuven Libraries. Downloaded on March 31,2021 at 08:08:01 UTC from IEEE Xplore. Restrictions apply.

of students has a team of three coaches for the non-technical aspects. The number of coaches per team may seem high, but usually only one takes an active role in the group, while the remaining two are left as backups in case of absence. Students work on the project every Monday afternoon throughout the year, and are supposed to organize themselves and ask relevant questions of the technical experts. The first author of this article is the technical expert for the wireless video link.

An In-Depth Overview of the Wireless Video Link Module

This section offers a more detailed description of the wireless video link module and elaborates on the results obtained in the current academic year. The DVB-S2 standard is briefly introduced, and the breadth of topics relevant to engineering curricula covered is discussed. We describe the design and implementation phases of the wireless video link module. Finally, we give an overview of the progress of the students and present their feedback on the learning process.

The DVB-S2 Standard and Its Educational Applications

The complete specification of the DVB-S2 standard can be found in [3], while we give a short overview of the main sub-blocks and discuss their application to engineering curricula.

The block diagram of the DVB-S2 transmitter can be found in Fig. 4. It consists of several sub-modules that can be indexed into four main categories:

1. Baseband (BB) processing blocks. These blocks format the incoming video data, and build the baseband frame by inserting the BB headers and adding the cyclic redundancy check (CRC). Finally, the whole stream is scrambled.
2. Forward error correction (FEC) blocks. These comprise Bose Chaudhuri (BCH) and low-density parity check (LDPC) error correction coding, as well as an interleaver.
3. Constellation mapping and physical layer (PL) framing blocks. They build the actual quadrature symbol stream.
4. Modulation and filtering block. In this block the RRC filter is applied. The data is formatted in a way the front-end can understand.

The input interface inside the BB processing blocks can be used for explaining the relevance of headers and control information. It shows how they allow the receiver to know how to decode the data stream. Moreover, it can be taken as an example of how communications systems are layered. The data stream from one technology, the camera output, is fragmented and packetized into upper layer frames for another technology.

The CRC encoder serves the students as an introduction to error detection codes, while the scrambler can explain the need for bias-removal techniques.

The FEC blocks extend the knowledge introduced by the CRC encoder. They can also be used to further strengthen the mathematical background of the students as they are based in polynomial division and several mathematical properties that allow one to detect if the received bitstream is corrupted by the channel.

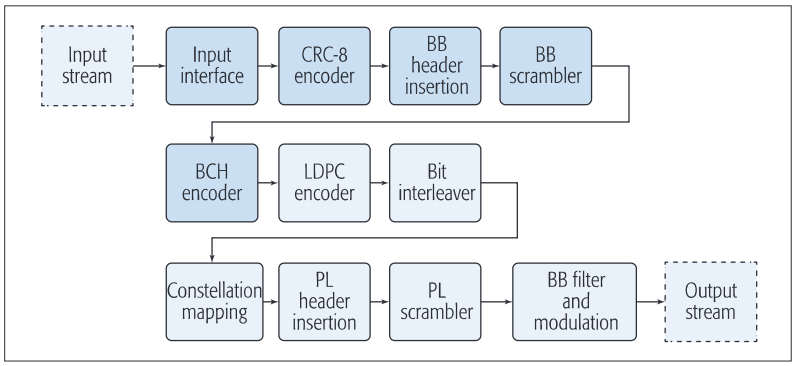

Figure 4. DVB-S2 transmitter block diagram. Shaded blocks are to be implemented by students during the project.

The constellation mapping blocks introduce the students to several modulation schemes, and their performance and robustness to errors.

The use of the final RRC filtering block in the educational process can be twofold: first, it can reinforce the digital signal processing (DSP) background of the students on digital filter design as well as offer a textbook illustration of the Nyquist pulse shaping criterion and the concept of matched filtering; second, it allows discussion about the spectral efficiency of the chosen modulation scheme and more high-level topics such as spectrum regulation, where such a filter is used to meet spectrum mask constraints.

As introduced before, students receive a partial implementation of the transmitter, starting from the LDPC encoder onward. This is done to specifically introduce or reinforce the topics covered by the BB processing and BCH blocks, which are taught in the remaining subjects studied during the academic year.

Design and Implementation of the Wireless Video Link

Design Phase: When the complete picture of the transmitter is clear, students need to design the BB processing and FEC blocks.

For the design of the BB processing an input interface, CRC-8 encoder, frame builder (BB header insertion), and scrambler need to be designed. In the ZyBo architecture the incoming MPEG-2 video stream is delivered in byte form, while the transmitter must process one bit per clock cycle. Therefore, the input interface must serialize this data stream and handle the synchronization of the transmitter with the camera to ask for a new byte of data only when the previous one has been processed and the remaining blocks are ready. Regarding the frame builder, the students need to choose the appropriate values of the header according to the application and build the BB frames, ensuring that the incoming video stream is fragmented to fit within the frame length. For the CRC-8 encoder, a linear feedback shift register (LFSR) must be designed in order to compute the checksum. This design can be reused for the scrambler, teaching the students the different applications of LFSRs.

For the design of the BCH encoding block, the standard describes the mathematical operation using polynomial division. Students need to

Authorized licensed use limited to: KU Leuven Libraries. Downloaded on March 31,2021 at 08:08:01 UTC from IEEE Xplore. Restrictions apply.

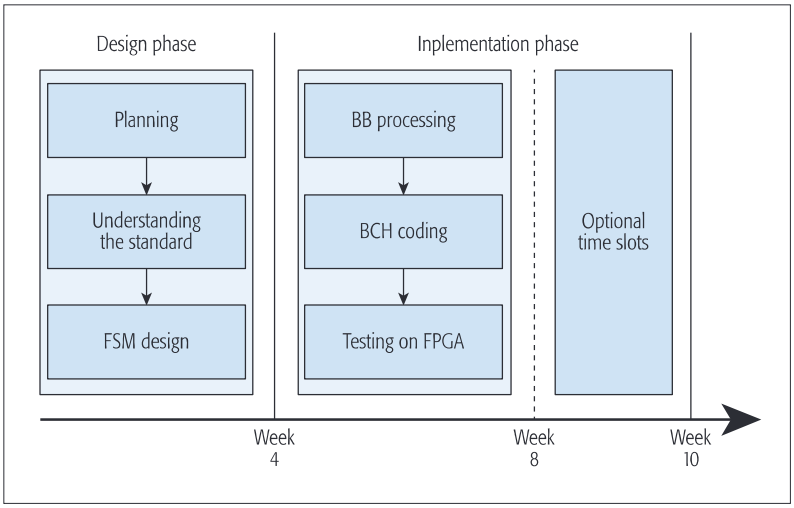

Figure 5. Expected schedule and milestones for the design and implementation phases of the wireless video link module. Extra time slots are reserved to optionally extend the functionalities of the transmitter, or to ensure that the baseline transmitter is working.

understand this operation and how the result is obtained, but implementing it this way would not be feasible in the FPGA. They need to again relate this operation with LFSR and propose a simple design for computing the BCH parity bits.

Implementation Phase: The implementation phase covers the actual coding of the selected sub-blocks, based on the proposed design. As testing and debugging a VHDL implementation with the off-the-shelf receiver results in a complex and binary process (it correctly receives and decodes the signal if there is no error in the implementation and completely fails to do so if there is), several standard compliance testbenches are provided to help locate faults in sub-blocks. This reinforces their critical thinking and teaches them the process of modular design, implementation, and debugging. The tests use custom test vectors containing the bitstreams before and after each block, so the code can be tested in pieces. Students are provided with tests for the BB processing with and without scrambler, the BCH encoder, the full FEC module, and finally the full transmitter, in which handshaking issues are tested. In order to test the handshaking, a `bit_ready` and `bit_valid` input and output port for each sub-block are defined. Each of the blocks coded by the students needs to process the signal only when the incoming bit is valid, while always waiting to deliver the output until the next block is ready to receive it. We generate random pulse sequences in these signals to ensure the proper synchronization of the whole transmitter. In this way, the students become aware of not only the theoretical concepts but also the practical challenges of a real-life system.

Once all tests are passed, the VHDL project is compiled and flashed into the FPGA for performance testing. In this last phase students need to also become familiar with the software for controlling the transmitter, and will finally be able to see the streaming of the camera.

Milestones and Deliverables: The wireless video link is expected to be designed and implemented during 10 weekly worksessions of 5 hours. There are three milestones in the global planning, which correspond to the three mandatory demo sessions planned for the first semester. The expected milestones are depicted in Fig. 5.

The design phase takes place during the first four weeks, and its main deliverable is a complete pseudocode design of the BB processing and BCH blocks using a finite state machine (FSM) approach. The complete DVB-S2 standard as well as material covering FSM design and VHDL coding is given to the students in this phase.

The implementation phase is divided into two sub-phases: another four weeks are planned for the VHDL coding, testing, and flashing into the FPGA, with two optional sessions at the end. These two sessions may be used for implementing adaptive coding and modulation (ACM) on the transmitter, characterizing its performance vs. signal-to-noise ratio (SNR) through a measurement campaign or simply finishing the project in case it is not done by then.

Student Progress and Feedback

Figure 6 shows the progress of the students during the first semester of the current academic year. The main conclusion that can be obtained is that the course represents a very challenging but achievable task for the students. Their performances are well spread, with all groups except one having a working VHDL code after the 10 sessions. However, the optional tasks (ACM and characterization of performance vs. SNR) were neglected by all groups.

Additionally, after the final week students are asked to submit a report providing feedback on the course. They have to reflect on the main difficulties found during the project and express their level of satisfaction with the results obtained and the learning process. A comparison with other project-based courses studied before as well as suggestions to improve the project for coming academic years are also asked.

Regarding the main difficulties, understanding the standard was the common one for all groups. However, while perceived as difficult, the standard did improve the overall satisfaction of the students with the learning process. The vast majority of them reported that, although challenging, they preferred using a real application as the driver for learning VHDL. This is contrary to other project-based learning courses they have done. Moreover, as the standard covers more topics than they have already studied, they appreciated being introduced to them. As a general suggestion, some initial sessions explaining the standard in more detail would have reduced the complexity of the project.

Summary of Educational Aspects

First, through the study of a real standard that is naturally decomposed in layers and functional blocks per layer, students learn how the divide and conquer approach is done in real technologies.

Second, students get their first hands-on experience in VHDL coding. After understanding the basics of VHDL, they learn to implement the individual blocks and to connect them considering the relevant interdependencies.

Of particular interest is the fact that students learn not only the relevant functionality for DVB-S2, but also how to implement this efficiently

Authorized licensed use limited to: KU Leuven Libraries. Downloaded on March 31,2021 at 08:08:01 UTC from IEEE Xplore. Restrictions apply.

in an FPGA. Here, they learn to understand the trade-off between functionality and implementation, and see how a standard is actually a representation of this trade-off.

Next to technical skills, students also improve their soft skills. Project management, planning, and communication also play a crucial role in EAGLE in order to meet the target deadlines. Students first have to decide how to divide the team of ten people over the different modules, how to meet regularly to discuss progress, and how to interact with the technical experts. It involves having discussions and making decisions with other teams about the functional decomposition of the tasks and the interfaces between the teams in order to guarantee smooth integration. It is very interesting to see that there is a strong correlation between technical progress and teamwork; that is, teams that have good planning, communication, and teamwork skills also outperform on the technical milestones.

Finally, students learn to write technical reports about the progress they made and the problems they encountered. This helps them to identify what is going wrong, narrow down the problem to the real bottleneck, and learn how to avoid it in the future.

Conclusions

In this course we used the DVB-S2 standard to teach students how to go from theory to practice and how to manage a complex engineering project.

There was some initial skepticism about the challenging nature of the project, and during the first two to four weeks students indeed felt overwhelmed. But it was remarkable to see that, after the initial reading of the standard, students conquered the most important insights, managed to get their VHDL code implemented, compiled, and tested, and delivered a working prototype. This disproves the idea that standards are too complex for Bachelor student education.

Several parts of the project were already given to the students. Depending on the curriculum and time allocated for the course, more blocks could be covered. If knowledge about PCB design or control of an analog front-end is needed, students could be asked to design a transmitter or focus more on the control of the front-end. If more focus is on baseband modulation theory, students could be asked to implement the extension for terrestrial broadcast, DVB-T2. It could also be interesting to compare the performance of both indoor and outdoor scenarios and understand the impact of multipath fading. If more focus is on the higher layers of the protocol stack, such as IP encapsulation or even the application layer, students can be asked to focus on the network stack. Finally, even application layer video compression could be optimized to meet the specific requirements of the EAGLE mission, and the students could be introduced to the concepts of cross-layer design.

As a conclusion and to emphasize the benefits of using a standard in a project-based learning course, being able to recognize a familiar platform (TV broadcasting) and learning the standard increases the motivation of the students to learn VHDL and code the transmitter, contrary to building a completely made up project that is not related to a real-world application.

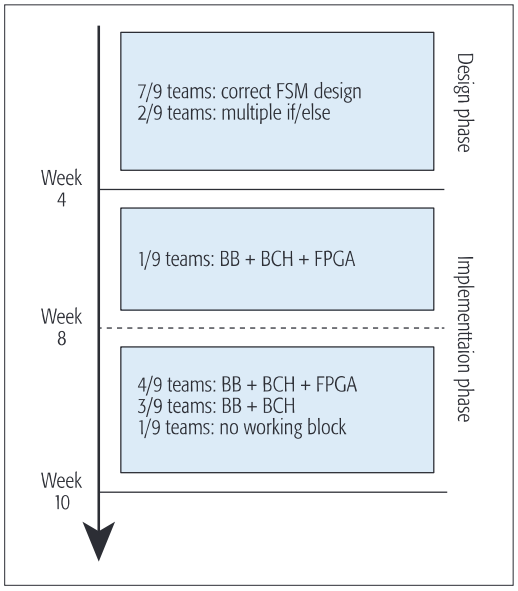

Figure 6. Student progress on the milestones. In the design phase all teams were able to make a plan and a correct VHDL design, although two teams did not follow an FSM approach and used multiple *if else* statements. In the implementation phase one team was able to finish in less than eight sessions, and four teams finished in ten sessions. Three teams had minor bugs in the FPGA implementation, while one team was not capable of having a working VHDL code.

References

[1] A. M. Agogino *et al.*, "Making Connections to Engineering During the First Two Years," *Proc. 1992 IEEE Frontiers in Education Conf.*

[2] M. J. Pavelich, B. M. Old, and R. L. Miller, "Real-World Problem Solving in Freshman/Sophomore Engineering," J. Gainen and E. Willemsen, Eds., *Fostering Student Success in Quantitative Gateway Courses*, Jossey-Bass, 1995.

[3] "Digital Video Broadcasting (DVB); Second Generation Framing Structure, Channel Coding and Modulation Systems for Broadcasting, Interactive Services, News Gathering and Other Broadband Satellite Applications (DVB-S2)," Nov. 2014, https://www.dvb.org/standards/dvb-s2, accessed Feb. 2017.

[4] Xilinx Inc. Zybo Zynq-7000 ARM/FPGA SoC Trainer Board, https://www.xilinx.com/products/boards-and-kits/1-4azfte.html, accessed Feb. 2017.

[5] C. L. Dym *et al.*, "Engineering Design Thinking, Teaching, and Learning," *J. Engineering Education*, vol. 94, 2005, pp. 103–20.

[6] A. J. Dutson *et al.*, "A Review of Literature on Teaching Design Through Project-Oriented Capstone Courses," *J. Engineering Education*, vol. 76, no. 1, 1997, pp. 17–28.

[7] B. Bennett *et al.*, "DVB-S2 Benefits for Military Broadcast Systems," *MILCOM 2005*, Atlantic City, NJ, 2005, vol. 1, pp. 148–54.

[8] KU Leuven, Project and Design EAGLE Course, https://onderwijsaanbod.kuleuven.be/syllabi/n/H01Q6CN.htm (in Dutch), accessed Feb. 2017.

[9] "Unidirectional Lightweight Encapsulation (ULE) for Transmission of IP Datagrams over an MPEG-2 Transport Stream (TS)," IETF RFC 4326, https://tools.ietf.org/html/rfc4326, accessed on Febr. 2017.

[10] A. Mayer *et al.*, "Analytical and Experimental IP Encapsulation Efficiency Comparison of GSE, MPE, and ULE over DVB-S2," *2007 Int'l. Wksp. Satellite and Space Commun.*, Salzburg, Austria, 2007, pp. 114–18.

[11] Zedboard, Xilinx Zynq-7000 ARM/FPGA SoC Based Development Board, http://zedboard.org/product/zedboard, accessed Feb. 2017.

[12] Raspberry Pi Camera Module v2, https://www.raspberrypi.org/products/camera-module-v2/, accessed Feb. 2017.

[13] TBS5980 QBOX CI DVB-S2 TV tuner USB, external TV tuner box for laptop and PC, http://www.tbsdtv.com/products/tbs5980-dvb-s2-ci-tv-tuner-usb.html, accessed Feb. 2017.

To emphasize the benefits of using a standard in a project-based learning course, being able to recognize a familiar platform (TV broadcasting) and learning the standard increases the motivation of the students to learn VHDL and code the transmitter, contrary to building a completely made up project that is not related to a real-world application.

Authorized licensed use limited to: KU Leuven Libraries. Downloaded on March 31,2021 at 08:08:01 UTC from IEEE Xplore. Restrictions apply.

Biographies

Yuri Murillo (yuri.murillo@esat.kuleuven.be) obtained his M.Sc. degree in telecommunication engineering from the Public University of Navarre (UPNA), Spain, in 2013. He wrote his Master's thesis at TU Delft, the Netherlands. From 2013 to 2015 he worked as an IT security consultant at GMV Secure-eSolutions and obtained a Master's degree in business administration from UAH, Madrid. He is currently pursuing a Ph.D. degree at KU Leuven, Belgium. His main research interests include optimization of dense wireless mesh networks and opportunistic networking protocols.

Bertold Van den Bergh (bertold.vandenbergh@esat.kuleuven.be) is a Ph.D. student in the Department of Electrical Engineering (ESAT), KU Leuven, Belgium. His research is focused on reliable communication systems for UAVs and other applications. Before starting his Ph.D. he obtained his Master's degree in microelectronics at KU Leuven in 2013, and did an internship at Flanders Make on wireless localization and tracking of mechatronic systems. He also worked at ETH Zurich on the Swarmix project.

Jona Beysens (jona.beysens@esat.kuleuven.be) obtained his B.Sc. and M.Sc. degrees in electrical engineering from KU Leuven in 2014 and 2016, respectively. After the first year of his Master's, he did an internship at National Chiao Tung University, Taiwan, designing a lightweight preprocessing algorithm for a deep learning convolutional neural network. Currently, he is pursuing a Ph.D. degree at KU Leuven, focusing on visible light communication for dense indoor networks and MIMO processing.

Alexander Bertrand [M'08, SM'16] received his Ph.D. degree in engineering sciences (electrical engineering) from KU Leuven in 2011. He is currently an assistant professor with the Electrical Engineering Department, KU Leuven. His research focus is on signal processing algorithm design for (distributed) sensor arrays.

Wim Dehaene has been a professor at KU Leuven since 2002. His main research interests are ultra low-power DSP and memories. Besides that, he also heads high-speed digital circuit research, mainly in the domain of digital RF transmitters and digital AD converters. He is also very active in research toward novel STEM education programs for the Flemish secondary school system. He is a member of the ESSCIRC TPC and served on the ISSCC TPC for many years.

Panagiotis Patrinos has been an assistant professor at ESAT, KU Leuven since 2015. During fall/winter 2014 he held a visiting assistant professor position at Stanford University. His current research interests are in structured optimization and model predictive control with a broad range of application areas including smart grids, water networks, aerospace, and machine learning.

Tinne Tuytelaars is a professor at the PSI research division of KU Leuven's ESAT.Her research interests lie in computer vision, and in particular object recognition, image representations, and transfer learning.

Ruth V. Sabariego is an associate professor with EnergyVille, ESAT, KU Leuven since October 2013. In October 2000, she joined the Department of Electrical Engineering and Computer Science, University of Liège, Belgium, where she received her Ph.D. degree in applied sciences in 2004 and stayed as a post-doctoral researcher until September 2013.

Marian Verhelst [S'01, M'09, SM'14] became an assistant professor at the Micro-Electronics and Sensors (MICAS) laboratories of ESAT, KU Leuven in 2012. Her research focuses on self-adaptive circuits and systems, embedded machine learning, and low-power sensing and processing for the Internet of Things. She is a member of both the ISSCC and DATE TPCs, and of the DATE and ISSCC Executive Committees. She was an Associate Editor for *TCAS-II* and currently is for *JSSC*.

Patrick Wambacq [SM] received his Master's degree in electrical engineering in 1980, and his Ph.D. degree in electrical engineering in 1985, both from KULeuven. He started as a lecturer at KU Leuven in 1985, becoming a professor in 1989 and a full professor in 2004. Initially he was interested in computer vision and industrial applications thereof, but since 1998 has been the head of the speech group at KU Leuven ESAT focusing on speech recognition and its applications.

Sofie Pollin [SM] (sofie.pollin@esat.kuleuven.be) obtained her Ph.D. degree at KU Leuven with honors in 2006. From 2006 to 2008 she continued her research on wireless communication, energy-efficient networks, cross-layer design, coexistence, and cognitive radio at the University of California, Berkeley. In November 2008 she returned to imec to become a principal scientist on the green radio team. Since 2012 she has been a tenured track assistant professor in ESAT, KU Leuven, Belgium. Her research centers around networked systems that require networks that are ever more dense, heterogeneous, battery powered, and spectrum constrained. She is a BAEF and Marie Curie fellow.

Authorized licensed use limited to: KU Leuven Libraries. Downloaded on March 31,2021 at 08:08:01 UTC from IEEE Xplore. Restrictions apply.