CONCEPT AND ACREDITATION OF NEW MASTER LEVEL COURSE SOFTWARE-DEFINED INSTRUMENTATION

Itasdi

Co-funded by the
Erasmus+ Programme
of the European Union

Факултет за електротехника и информациски технологии, Скопје 2019 год.

New master level course: Software-defined instrumentation

1. Introduction

One of the challenges that engineers face nowadays is the increased complexity of the devices they need to design and test. As devices continue to become more complex and include more disparate technologies, test systems must become more flexible. The only way to accomplish this is through software-defined instrumentation, which helps engineers develop scalable, high-performance test systems. Next-generation test systems must be flexible enough to support the wide variety of tests that differ among convergent products and they must be scalable enough to accommodate a larger number of tests as new measurement functionality is required. Software-defined instrumentation is the essential differentiator for meeting this test challenge. The mainstream adoption of software-defined modular instruments in automated validation and production test applications is confirmation of this trend.

The functionality of a modular instrument is characterized through user-defined software residing on the host personal computer (PC) instead of on the instrument. The role of software in modular instrumentation cannot be overstated. With a software-defined modular instrumentation system, engineers can quickly adapt to changing test needs. Through software, they can program a modular instrumentation system to function as one user-defined instrument using built-in shared clocks and triggers. On the other hand, software-defined instrumentation hasn't been deeply threated and systematically organized in literature so far and students are lacking of knowledge in this field. This course aims to fill this gap by introducing a new master level course into student's curriculum.

2. Course description

Software-defined instrumentation aims to be a new University course at ss. Cyril and Methodius University of Skopje, Faculty of electrical engineering and information technologies in the next master program accreditation cycle in 2021. The students will be able to participate in this course in their first semester of master studies at the study program Metrology and quality management. The usual number of participants is around 10 students per year dealing mainly with electronics and automation. Prior this academic course, students gain basic knowledge of measurement science and technology and they are skilled to understand and interpret the concept of uncertainty in measurements.

3. Identification of opportunities

Usually instrumentation manufacturers provide specific functions to given architecture and fixed interfaces for measuring devices, and thus limit the application domain of these devices. In actual use much time is required for adjusting the measuring range and for storing and documenting the results. The advent of microprocessors in the measurement and instrumentation fields produced rapid modifications of measuring device technology, soon followed by the appearance of computer-based software-defined measurement systems. Using the design software or similar software inside the instruments allows designers to create software-defined instrument applications and also offers many benefits in terms of a design and verification flow.

Typically, a design and verification flow begins with a solution used to put together the system and circuit designs. At this stage of the design flow, test equipment is not yet involved since it is still a design-only phase. Simulation signal sources and signal measurements are used in the software to begin designing the system and circuits. For emerging technologies, signal sources and measurements could be available as predefined simulation test benches, which are pre-configured to perform key simulation measurements. When the design phase is complete and device under test (DUT) hardware returns from fabrication, early research and development (R&D) verification testing can begin. If off-the-shelf instrumentation solutions are not readily available for the given signal format, then the software can be combined with test instrumentation to provide an early R&D test solution. In such case, it may also be useful to combine the software with test instrumentation to create custom R&D test solutions. There have been numerous publications written on concepts like Connected Solutions to extend the instrumentation's functionality for R&D applications [1-6]. Most of the work-to-date has been implemented in the form of an external PC (such as a laptop) connected to the test instrumentation to perform the signal generation and signal analysis. However, once the Connected Solution has been created, the designer might take this a step further by installing the design software inside the instrument solution. This is somewhat dependent on the instrument platform, and whether it can support installing and running the design software or similar software. After the software is installed on the instrument, it might also be configured to run the measurement automatically by pressing a button on the instrument's front panel or by pressing a button in the instrument's application software. This effectively creates a software-defined instrument, using the design software to define the signal source and measurement.

From a technical viewpoint, the idea of a "software-defined instrument" is not a new one. For example, (in the early 2000's) three critical factors drove the transition of the modern signal analyzer from a predominantly analog instrument to a digital one. First, the increasing integration of multiple radios onto singular products (such as the smartphone) drove the requirement for radio frequency (RF) signal analyzers to test every wireless standard imaginable. Second, the continued evolution of existing standards drove the requirement for RF signal analyzers to be more easily upgradable. Finally, continued innovation in the embedded processing world introduced a new generation of microprocessors and field programmable gate arrays (FPGAs) that were capable of handling the signal processing challenges of RF instruments.

In the software-defined RF instrument, the core measurement functions are performed with a CPU instead of using traditional approaches. One of the clear benefits of this architecture is that software-defined instruments are able to support composite measurements which are performed in parallel. Even today, the software-defined instrument has many benefits over the traditional instrument approach. First, since measurements are performed in software, the instrument can be easily upgraded to analyze new signal types or signal formats. However, perhaps one of the most understated benefits of the software-defined instrument is the advantages in measurement speed over the traditional approaches to instrumentation. As we look to the future of software-defined instrumentation, two critical trends that will continue to drive innovation in measurement science are the needs for 1) greater signal processing capabilities and 2) productive measurement abstraction. Thus, over the next decade, the term "software-defined instrument" will continue to take on new meaning as engineers use software as the primary graphical system tool used to configure their instrumentation.

The course Software-defined instrumentation will be covered by a large teaching material which demands a good knowledge of other fields in electrical engineering (electrical measurements, instrumentation, communication techniques, programming, etc.). Thereby, students will be obliged to analyze a lot of different engineering approaches and solutions. Such efforts are more effective if supported by practical implementations and a possibility for experimentation work. However, having in mind that the laboratory exercises are limited in time and resources, such approach is often a challenge. Our experience over the years shows that students are lacking of user friendly interactive tools to support the motivation of autonomous experimentation work at home. This would increase the effective time used to design/test/implement a given solution and increase the quality of learning, especially when it comes to practical implementations.

On the other hand the students at Faculty of electrical engineering and information technologies (but also at another Universities) are familiar with the concept of Virtual Instrumentation as a part of the LabVIEW Academy program in the lower semesters. The availability of such tools and knowledge can compensate for the weaknesses stated above. Namely, the virtual instrumentation (VI) can be used to implement **VI experimentation package** in the area of Software-defined instrumentation and thereby support the autonomous experimentation work at home.

Another motivation to implement the Virtual Instrumentation concept is the student's satisfaction to use the previous gained knowledge in reality, and to exploit the availability of the licensed LabVIEW Academy software package. Besides the improvements by the innovation of the education process, this application also aims to provide a course program improvement by updating it with the state-of-the-art approaches in the field of software-defined instrumentation. The foreseen program improvements are elaborated in chapter 5.

4. Proposed course contents

The course Software-defined instrumentation is conceived with the following main topics (at this stage):

- Software-defined instrumentation architecture
  - Sensor interfaces
  - Programmable instrumentation
  - Database interface
  - Processing module
  - Presentation and control
  - Functional integration
- Distributed instrumentation
  - Private networks
  - Internet
  - Cellular networks
  - Distributed integration
- Tools and platforms
  - Hardware platforms and operating systems
  - Development environments
- Measurement uncertainty in software-defined instrumentation
- Applications of software-defined instrumentation

5. Implementation of a virtual instrumentation experimentation package and laboratory exercises

The virtual instrumentation experimentation package is conceived to contain eight virtual instruments that cover the majority of the course program topics. The virtual instruments will be license free (education license) and hardware independent executable programs intended for individual experimental work in home environment. They will be divided into four main groups: programmable instrumentation; communication interfaces; distributed instrumentation; and measurement uncertainty.

With the laboratory exercises, the students are in the position to perform a practical measurements and experimentation, ant thereby test the theoretical material covered with the course. They are often considered as a critical part of the education process, where students are trained to apply the gained theoretical knowledge in reality. It is therefore expected to develop new laboratory exercises with all aspects highlighted in the previous sections. Moreover, the software-defined instrumentation is subject of continual technological improvement followed by a new communication/interfacing techniques. In that regard, it is planned to design laboratory exercise models in the following areas: programmable instrumentation, virtual instrumentation, distance laboratory, communication interfaces, signal processing, data logging, and measurement uncertainty.

6. Oficial acreditation document

Прилог бр. 3	**Предметна програма од втор циклус на студии**				
1.	Наслов на наставниот предмет	**Софтверски-дефинирана и виртуелна инструментација**			
2.	Код	ФЕИТ03008			
3.	Студиска програма	Метрологија и менаџмент на квалитет (ММК)			
4.	Организатор на студиската програма (единица, односно институт, катедра, оддел)	Факултет за електротехника и информациски технологии			
5.	Степен (прв, втор, трет циклус)	Втор циклус			
6.	Академска година/семестар	V/9	7.	Број на ЕКТС кредити	6
8.	Наставник	Вон. проф. д-р Живко Коколански, Доц. д-р Томислав Шуминоски			
9.	Предуслов за запишување на предметот	Нема			
10.	Цели на предметната програма (компетенции): Со успешно положување на овој предмет, кандидатот ја совладува софтверски-дефинираната виртуелна инструментација и компјутерски мрежни системи. Се стекнуваат знаења за програмабилна инструментација, аквизиција на податоци, виртуелна инструментација и компјутерски мрежни системи.				
11.	Содржина на предметната програма: Вовед во софтверски-дефинирана инструментација и нивни архитектури. Основни комуникациски интерфејси на софтверски-дефинираната инструментација и нивна примена. Напредни аспекти на виртуелна инструментација. Професионални архитектури на виртуелната инструментација. Дистрибуирани мерења, телементрија и пренос на податоци. Споредба на SDN и традиционални мрежи, предности и цели на SDN мрежите, архитектура на SDN мрежите. OpenFlow протокол, предности и цели на NFV мрежите, архитектура на NFV мрежите, споредба на SDN и NFV, SDN & NFV во паметни градови (Smart Cities) и паметни услуги. Паметни уреди, поврзани уреди, IoT уреди, паметна мрежа, паметни мерила, паметни згради (Smart Building), паметни системи за транспорт. Примери за примена на SDN во 5G и сателитски мрежи.				
12.	Методи на учење:				
13.	Вкупен расположлив фонд на време	180			
14.	Распределба на расположливото време	3+0+0+3			
15.	Форми на наставните активности	15.1	Предавања-теоретска настава	45 часови	
		15.2	Вежби (лабораториски, аудиториски), семинари, тимска работа	/	
16.	Други форми на активности	16.1	Проектни задачи	45 часови	
		16.2	Самостојни зададчи	/	
		16.3	Домашно учење	90 часови	
17.	Начин на оценување				
	17.1	Тестови	30 бодови		
	17.2	Семинарска работа/проект (презентација: писмена и усна)	50 бодови		
	17.3	Активност и учество	20 бодови		
18.	Критериуми за оценување (бодови/оценка)	до 50 бода	5 (пет) (F)		
		од 51 до 60 бода	6 (шест) (E)		
		од 61 до 70 бода	7 (седум) (D)		
		од 71 до 80 бода	8 (осум) (C)		
		од 81 до 90 бода	9 (девет) (B)		
		од 91 до 100 бода	10 (десет) (A)		
19.	Услов за потпис и полагање на завршен испит	60% успех од сите предиспитни активности			
20.	Јазик на кој се изведува наставата	Македонски/Англиски			
21.	Метод на следење на квалитетот на наставата	Самоевалуација			

22.	**Литература**					
	22.1.	**За должителна Литература**				
		Ред. Број	Автор	Наслов	Издавач	Година
		1.	G. Johnson, R. Jennings	LabVIEW Graphical Programming	Mc Graw-Hill	2006
		2.	Mike Tooley	PC Based Instrumentation and Control	Elsevier, 3 Ed.	2005
		3.	National Instruments Corp.	LabVIEW Core 1, 2, 3 - Course Manual	National Instruments Corp.	2009
	22.2.	**Дополнителна Литература**				
		Ред. број	Автор	Наслов	Издавач	Година
		1.				
		2.				
		3.				

7. Refferences

[1] T. Helaly, N. Adnani “A fourth category of software-defined instrumentation for wireless test”, IEEE Instrumentation & Measurement Magazine, Vol. 20, Iss. 4, Aug. 2017

[2] G. Jue and S. Ferguson, “RF And Digital Tests Unite Against BER,” Wireless Systems Design Magazine, Nov. 2004

[3] “Connected Simulation and Test Solutions Using the Advanced Design System,” Agilent Technologies, Application Note Number 1394

[4] G. Jue, “3GPP W-CDMA Systems: Design and Test,” IEEE Microwave Magazine, June 2002, pp 56-64

[5] B. Zarlingo, K. Kalbasi and G. Jue, “Flexible Digital Demodulation/Integrating Simulation Software with Measurement Hardware,” IEEE Autotestcon 2002

[6] D. Leiss, “Combined Virtual and Physical Hardware Performance Analysis,” IEEE Autotestcon 2002

Dr. Zivko Kokolanski, Associate Professor at FEEIT