Internet of Things
Course Syllabus

This course contributes to the requirements for the Degree of MSc in Computer Science

Title of the Academic Program	Master's Degree Programs in English "Digital intelligent control systems"
Type of the course	core /mandatory
Course period	1 semesters from October 1st till February 1st (18 weeks)
Study credits	3 ECTS credits
Duration	108 hours
Language of instruction	English
Academic requirements	– BSc degree in Computer Science or equivalent (transcript of records), – good command of English (certificate or other official document) **Prerequisites:** – **base knowledge of** digital electronics, programming skills.

Course Description

"Internet of Things" is a core course.

The IoT course provides a comprehensive illustration of emerging Internet of Things (IoT) systems, by focusing on both methodological and applicative/industrial aspects, and including both theory classes and laboratorial activities.

The course program follows a data-centric approach, by presenting the data-path from the sensor acquisition till the "data valorization" through machine-learning and analytics techniques, and considering all the intermediate phases of data communication via emerging Machine-to-Machine (M2M) communication standards, streaming, storage on public/private clouds, and integration through Web semantic approaches.

After a short introduction to pervasive computing and the emerging applications (e.g. Industry 4.0, demotics, intelligent transportation systems, smart energy, etc.) the course will provide an exhaustive illustration of the enabling components of IoT systems. These include: wireless standards for short and long M2M communication (e.g. BLE, 6LoWPAN, LoRa, NB-IoT, …) among sensor edge-devices, languages and boards for the development of prototypal WSAN, WPAN, WLAN sensor networks (e.g. Arduino, STM32 Nucleo, RaspberryPI, …), and IoT network stacks and protocols (e.g. CoAP, MQTT, AMQP, …) From the data transmission, we move to architectures and tools for data storage and processing, including sensor time-series database (e.g. InfluxDB) and fog/cloud approaches and services (e.g. Amazon AWS IoT, ThingSpeak). Finally, the course will explore techniques for the valorisation of sensor data and knowledge extraction from large IoT deployments and crowd-sensing environments, through the application of machine-learning and data-analytics techniques (e.g. time-series prediction, classification, clustering, anomaly detection, etc.). Current open issues, like scalability, data interoperability (with a special focus on Web-of Things approaches) and energy-efficiency will be largely discussed.

Special Features of the Course

The course provides an opportunity for students to work individually using electronic and test equipment. The student will be able to go all the way from the emergence of an idea to project implementation using Arduino hardware or its analogs.

The aim of the course is to provide students with knowledge and skills of developing, test, debug a digital system with networking protocols using computer aided design tools.

Course Objectives

- to familiarize students with principles functioning and architectures modern devices, capable of IoT functionality;
- to acquaint students with the interfaces between sensors, network devices and ARM Cortex devices;
- to teach students to use computer aided tools and techniques for designing IoT devices.

Learning Outcomes of the Course

By the end of the course, students will know:

- structural features and applications of the most common IoT compatible device families;
- general knowledge about digital interfaces for sensors;
- basics of programming for ARM Cortex devices (Use libraries);
- general principles and approaches to debugging and verification of embedded systems.

By the end of the course, students will be able to:

- choose the most appropriate class of embedded system to solve a problem;
- perform modeling, optimizing and debugging for a IoT projects.

By the end of the course, students will possess:

- the necessary skills to design IoT compatible embedded devices and systems using computer aided tools and techniques.

Course (module) Structure

Learning Activities	Hours
Lectures	18
Practice sessions / Seminars	18
Self-study Assignments	72
Total study hours	**108**

Detailed Schedule

Week	Lectures	Seminars/ Assignments	Hours Lec/Lab/HA
Semester 1			
1-2	A Practical Introduction to ARM® Cortex®, compiling, downloading, and running simple programs on an evaluation board.	CAD tools for STM - based design. STM32CubeMX, AC6 System Workbench (SW4STM32) IDE and STM Studio – review. STM32F4 – evolution board review. ARM –	2/4/8

		Processors and STM-Library – review (HAL/LL).	
3-4	CAD tools and libraries (HAL/LL) for STM32-based design. C Language Programming, introduce to writing programs in C, a popular programming language among embedded-system developers.	Hello world by STM. Pulse generator by LED and Switches.	2/2/8
5-7	Programming I/O, investigates some of the functions that configure I/O devices, gaining an understanding of what is involved in writing I/O interfaces for other targets. Digital interfaces. Serial interface, UART. I2C, SPI, CAN interfaces – review.	Digital ports GPIO, Timers and PWM – mode. UART, I2C-Interface.	2/4/8
8-10	Real-Time Signal Processing, introducing to Digital Signal Processing (DSP) and review the ARM Cortex M4 instruction set support for DSP applications.	DMA application using the codec, followed by designing a low-pass filter.	4/2/8
11-12	Real-Time Embedded Systems, writing a multithreaded program using flags for communication and ensuring mutual exclusion when accessing shared resources.	Programing of Real-Time Embedded Systems.	4/2/18
13-18	Session-layer Protocols: MQTT and CoAP. HTTP-based M2M and the Web of Things. The W3C Web of Things.	Final complex project. IoT system prototype.	4/4/20
	18	**18**	**108**

Course Instructor(s) and Tutor(s), Contact information

Dmitry A. Schwetz,
Ph.D. in Mathematical and software support for computers, complexes and computer networks, Lecturer, School of Space and Information Technologies, Siberian Federal University.

e-mail: DSchwetz@sfu-kras.ru
Google Scholar page:
https://scholar.google.com/citations?user=MWXrXhwAAAAJ
Additional information is available at:
http://structure.sfu-kras.ru/node/4244

Assessment

Assessment strategy	Points, max	Evaluation criteria
Tests	10	Test questions for lectures in the e-course
Lab works	40	Lab report
Individual Project	40	Electrical schematics, code, report on the project, presenting the project

Final exam (mandatory)	10	2 questions and a practical task that require preparatory reading and knowledge of the concepts explained

Grading policy for final assessment is:

- A (excellent work) 91–100 points
- B (above average) 81–90 points
- C (average) 71–80 points
- D (below average) 50–70 points
- F (failed) < 50 points

Attendance Policy

Students are expected to attend classes regularly. In case of missing an in-lab activity a student should perform additional work submitted to the instructor within a week after a class was missed.

Every topic involves an assignment. A written report on the assignment should be submitted within two weeks from the moment students received a list of problems. The final mark will rely on the same grading policy as for the final exam.

Web page of the course

Course materials and required reading materials are available on the course webpage “Internet of Things”. The webpage is available through the SibFU E-learning portal www.e.sfu-kras.ru. You must be logged in to access this course. https://e.sfu-kras.ru/course/view.php?id=31565

Core reading

1. Simone Cirani, Gianluigi Ferrari, Marco Picone, Luca Veltri. Internet of Things: Architectures, Protocols and Standards. Wiley, 2019. p.383. ISBN 9781119359678.

2. Perry Xiao. Designing Embedded Systems and the Internet of Things (IoT) with the ARM® Mbed™. Wiley, 2019. p.316. ISBN 9781119363996.

3. Getting Started with Atmel Studio 7. User Guide DS-50002712A. © 2018 Microchip Technology Inc. Access for free: https://ww1.microchip.com/downloads/en/DeviceDoc/Getting-Started-with-Atmel-Studio7.pdf

Facilities, Equipment and Software

Software:

Atmel Studio 7, Free, no license required;
Proteus Virtual System Modelling (VSM);
Microsoft Office®.

Laboratory equipment:

Atmel STK 500, STK501, STK600 Evolution board;
Arduino Uno Evolution board;
Sensors, actuators and connectors – bag.

Control, testing and measuring equipment:

Digital oscilloscopes PV6501, GW Instek GDS-8205, Tektronix TPS 2024;
Multimetr ABM-4307;
Signal generator GW Instek SFG-2010.