# C and RTOS for Small Embedded Systems
# EE-CES 3CR

Instructor: Dr. Aleksander Malinowski http://gdansk.bradley.edu/olekmali/

## Course Description

Advanced programming of small microprocessor-based systems using high-level programming languages applied to real situations; data acquisition, control, communication, small real time operating systems. Software development for devices from a family of microcontrollers that are relevant to industrial applications.

## Prerequisites by topics

Data structures in C or C++ (EE221) and microprocessor systems (EE365); Currently, the course builds on familiarity of architecture and programming of 8051 family of processors gained in EE365.

## Textbooks and/or other required material

1. *C and the 8051, 4th Ed.*, by Thomas W. Schultz (see also http://www.candthe8051.com/)
2. Silicon Labs 8051 Data Sheets
3. FreeRTOS (see also documentation at http://www.freertos.org/)
4. Additional materials posted on the course Web site and Internet

## Long Distance Education Students

This course heavily utilizes Embedded and Network Laboratory (http://gdansk.bradley.edu/olekmali/projects/embedded/). Students who do not live in the area would need to purchase Silicon Labs C8051F120 Development Kit ($100) or Ethernet Development Kit (https://www.silabs.com/products/mcu/Pages/EmbeddedEthernetDK.aspx) ($120) and make arrangements for alternative homework/projects to those that utilize different boards.

## Course Objectives

1. Learn how to program in a high level programming language for embedded applications
2. Learn the hardware of the 8051 family of microcontrollers
3. Develop the mindset of multitasking and interrupt driven programming
4. Develop the mindset of event-driven data communication using UART/RS232 and USB
5. Develop an appreciation for efficiency of use of computing resources

## Topics Covered

1. C and hardware: Variables, Operators, Branching and Looping, Functions, Arrays, Structures and Pointers
2. Modular Programming: C Modules, Assembly Modules, Scope of Variables and Functions, Mixing Languages
3. Hardware: Basics and Assembly Language, 8051 Instruction Summary, Clock Cycles and Software Delays
4. Hardware: Memory, Ports and Expansion, Interrupts, Counters and Timers
5. RTOS: Multitasking, Basic and Commercial RTOS, Scheduler, co-routines, cooperative and preemptive task switching, inter-task communication using queues and semaphores, priorities and deadlocks
6. Algorithms: User Input/Output, Number Conversion, ASCII, Parsing, Random Numbers, Scaling, Tables, Look-up Tables, Linear interpolation, Floating Point Variables
7. Algorithms: Time, Measuring Frequency or Period, Maintaining a Calendar
8. Algorithms: Self-calibration of Sensors; Digital Signal Processing; Smoothing and Averaging; Control Algorithms; Adaptive Control (Self-tuning);
9. Networks: Serial Communication, UART Fundamentals, Examples, Software-only vs. Serial Driver Hardware EIA/TIA-232 (RS-232) Specification, Other Serial Protocols
10. Algorithms: Buffers (Single-use, Cyclic FIFO/Ring)
11. Hardware: Digital Input with TTL, Switches and Buttons, Keypads, Keyboards and Digital Output LED, LCD
12. Hardware: Analog Input with A/D Conversion, and Analog Output with D/A Conversion