ECE 471/571 Real-time Operating Systems – 3CR

Instructor: Dr. Aleksander Malinowski http://olek.matthewm.com.pl/

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 is relevant to industrial applications.

Prerequisites by topics

Computer programming in C or C++ (ECE103) and microprocessors (ECE205)

Textbooks and/or other required material

1. *C and the 8051, 4th Ed.,* by Thomas W. Schultz / 978-1581124590 / (see also http://www.candthe8051.com/)
2. FreeRTOS (see also documentation at http://www.freertos.org/)
3. Silicon Labs Data Sheets for the 8051F120 mixed signal processor development board – "the fastest 8-bit MCU" (http://www.silabs.com/products/mcu/8-bit/c8051f12x-f13x/Pages/c8051f12x-f13x.aspx)
4. Additional materials posted on the course Web site and on the Internet

Course Objectives

1. Improve programming skills in a high-level programming language for embedded applications
2. Learn the hardware of the 8051 family of microcontrollers (CISC)
3. Compare the hardware of the 8051 family with the hardware of the 8-bit Atmel family of microcontrollers
4. Learn the core features of a small RTOS including setup, task management, and task communication
5. Develop the mindset of multitasking and interrupt-driven programming
6. Develop the mindset of event-driven data communication using UART/RS232, CAN bus, and USB
7. Develop an appreciation for efficiency of use of computing resources

Topics Covered

1. C90 and C99: Variables, Operators, Branching and Looping, Functions, Arrays, Structures and Pointers
2. Modular Programming: C Modules, Assembly Inline, Scope of Variables and Functions
3. Hardware: Memory, Ports and Expansion, Programmable Clock Circuity, Interrupts, Counters and Timers
4. RTOS: Multitasking, Basic and Commercial RTOS, Scheduler, co-routines, cooperative and preemptive task switching, intertask communication using queues and semaphores, priorities and deadlocks
5. Algorithms: User Input/Output, Number Conversion, ASCII, Parsing, Random Numbers, Scaling, Tables, Lookup Tables, Linear interpolation, Limited Floating Point Variables
6. Algorithms: Time, Measuring Frequency or Period, Maintaining a Calendar
7. Algorithms: Digital Signal Processing; Smoothing and Averaging; Control Algorithms; Adaptive Control (Self-tuning);
8. Networks: Serial Communication, UART Fundamentals, Examples, Software-only vs. Serial Driver Hardware EIA/TIA-232 (RS-232) Specification, Other Serial Protocols
9. Algorithms: Buffers (Single-use, Cyclic FIFO/Ring)
10. Hardware: Digital Input with TTL, Switches and Buttons, Keypads, Keyboards and Digital Output LED, LCD
11. Hardware: Analog Input with A/D Conversion, and Analog Output with D/A Conversion
12. MISRA - Automotive Industry Consortium Coding Rules