

# Syllabus

## Course Description

Hardware implementation of a microcomputer architecture using modern microprocessors and related integrated circuits: clock subsystem, bus drivers, map decoders, R/W memory, ROM, serial and parallel I/O, DMA, interrupts.

## Course Catalog Description

Microcomputer architecture design describing the system bus, memory subsystems and peripherals. Unidirectional and bidirectional system bus; SRAM, SDRAM and FLASH memories and their bus interfaces; DMA, interrupt controller, transmitter/receiver, timers, display adapter, A/D and D/A converters and other system peripherals and their interfaces with system bus.

## Planned classes

### Class #1: Introduction

Introduction to the SJ2 board, and development environment

- Discuss class expectations
- Introduction to SRAM and Flash memory
- Mail out the boards
- Setup slack for class collaboration

### Class #2: Board Software Introduction

- [SJ2 project code structure](#)
- [Schematic](#) review
- Bootloader, Reset and NMI (NXP Boot) pin
- [SJ2 board startup](#)
- [LPC User Manual pdf](#) brief review
- Clock system
  - Discuss the role of a PLL
  - Peripheral clock divider
- How `printf` really works

### Class #3: Hands-on experiments

- `lpc40xx.h` memory map review and LPC User manual review
- Bit masking, LPC provided memory map

- LED and switch interface
- SJ2 unit-test framework

## Class #4: DMA

- Introduction to the peripheral
- Volatile keyword
- Lab assignment: Setup memory-to-memory transfers

## Class #5: Midterm

- First 60 minutes: Review session and Timer Peripheral walkthrough
- Review session, followed by the exam

## Class #6: Timer

- Setup HW timer that rolls over each second
- Lab assignment: Build an API to create a precise delay of nanoseconds

## Class #7: DMA to GPIO

- Lab assignment: Setup timer to trigger for the DMA
- Transfer a block of memory to the GPIO memory

## Class #8: UART driven by GPIO

- UART communication bus
- Lab assignment: Use the timer API to delay by nanoseconds

## Class #9: Memory

- SRAM layout
- Flash layout
- Startup and linker script
- EEPROM
- Lab assignment: Write a "peripheral driver"

## Class #10: Future facing knowledge: FreeRTOS

- Basics of an RTOS
- Stack pointer
- Lab assignment: Create multiple tasks
- Final exam review

## Last class

- Final examination
  - Goodbyes
-

Revision #16

Created 4 years ago by [Preet Kang](#)

Updated 1 month ago by [Preet Kang](#)