

Microprocessors & Interfacing

Review

Lecturer : Annie Guo

Lecture Overview

- Course review
- About the final exam

What have we learned?

- Basics of microprocessor systems
- AVR and AVR programming
- Interfacing
- Microcontroller application development

Basics of Microprocessors

- Microprocessors are digital systems
 - Can be programmed to perform varied functions on different information.
- Information can be encoded
 - Basically with binary digits
 - For example
 - Numbers in binary code, 2's complement code, ...
 - Characters in ASCII code

Basics of Microprocessors

- Five basic components of hardware computer systems
 - Datapath, control unit, memory, input/output devices
- Microprocessor
 - Datapath + control unit → on one chip
- Microcontroller
 - Microprocessor + memory + peripherals → on one chip
- Computer application systems
 - Hardware + software
- ISA
 - Interface between hardware and software

ISA

- Instruction set architecture
 - Things that assembly programmers can use and should know
 - Including
 - Instruction set
 - Supported native data type
 - Registers
 - What kinds of registers are available?
 - How to use them?
 - Memory models
 - How are instructions or data stored?
 - How can memory be accessed?
 - Interrupts schemes
 - What kinds of interrupts are available?
 - How to use them?

AVR and AVR Programming

- Assembly programming
 - can be constructed of
 - Sequential operations, if-then-else, loops
 - Function implementation
 - Stack and stack frames
 - Functions and function calls
- Data and data structure implementation
 - Constants, variables
 - Integers, characters, arrays, ...
 - Bits
 - Control bits

Interfacing

- I/O devices
 - Input devices: switch, push button, keypad
 - Output devices: LED bar, LCD
- Basic interaction approaches
 - Polling and Interrupt
- Basic communication types
 - Parallel and serial input/output
- Handling analog input/outputs
 - ADC, DAC, PWM

Project Development

- You need to
 - Find out what resources are available
 - How to use available resources
 - Develop strategies and design for the project
 - Hardware and software
 - Test and debug

What can we do?

- Not only know what, but how, and more importantly why.
- For the practical problems encountered in the future work or re-learning process, you can
 - identify the basic knowledge needed and find the possible solutions.
- Maintain (existing) microprocessor application systems
 - Problem debugging
 - System enhancement
- Design a microprocessor application system
- Be able to proceed to extended/advanced areas for breath/in-depth study

Where can we go now?

- Proceed to advanced computing courses
 - Compiler
 - For large software development, automatic machine code generation
 - Computer architecture
 - How to improve the processor and computer hardware system?
 - Operating system
 - How to make good and efficient use of the computer hardware system
 - Networking
 - How to handle complicated communications and make communication efficient?
 - Embedded system design
 - How to develop an efficient processor for a specific application
 - How to design and implement an embedded system with microcontrollers, microprocessors and other specific designed hardware components.

About Final Exam

- Moodle Online
- Date: Wed., Dec. 9
- Exam open time: 13:00 (Sydney time)
- Exam close time: 16:00 (Sydney time)
- Duration: 2 hours
- Two types of questions
 - 15 multiple choice questions
 - For some questions, multiple answers are allowed.
 - 2 programing and design questions

About Final Exam (cont.)

- Materials will not be covered in the exam
 - Slides marked with *
 - Extended topic
 - AVR ADC
- Sample questions will be posted in Moodle
 - <https://moodle.telt.unsw.edu.au/mod/quiz/view.php?id=3338423>

Thank You!

myExperience (CATEI) Survey

- Please participate
- If you have some feedback not covered by the Survey, please send your comments to me.
- Your feedback is much appreciated and will be considered for future improvement.