ECED3204: Microprocessor Course Review

Jason J. Gu
Department of Electrical and Computer Engineering
Dalhousie University

Outline

1. Introduction to Microcontroller (*)
2. Introduction to AVR Microcontroller (***)
3. AVR Assembly Language (***)
4. Hardware and Software Tools (*)
5. Advanced Assembly Programming (***)
6. C Language Programming (**)
7. System Clock (*)
8. Parallel I/O (***)

Outline (cont'd.)

9. Advanced I/O (***)
10. Interrupts (***)
11. Timer Functions (***)
12. USART (***)
13. The SPI Function (***)
14. Two-Wire Interface (TWI) (***)
15. Analog-to-Digital Converter (***)

I: Introduction to Microcontroller

- Characteristics of a mainframe computer
- Characteristics of a minicomputer
- Characteristics of a microcomputer
- Characteristics of a supercomputer
- Characteristics of a microprocessor
- Characteristics of a microcontroller
- Characteristics of an embedded system
- Structure of a computer hardware
- Memory technologies

II. Introduction to AVR Microcontroller

- **Key Concept**
  - AVR memory space configuration
  - AVR CPU registers
  - X, Y, and Z registers
  - Addressing modes
  - Data memory and program memory
  - Addition and subtraction operation
- **Key Computational Skills: Be able to**
  - Access data memory and program memory
  - Perform the addition and subtraction

Electrical and Computer Engineering
Dalhousie University

III. AVR Assembly Language

- **Key Concept**
  - Use directives to define constants, reserve memory locations, define macros, set location counter, and so on
  - Perform arithmetic and logic operations
  - Program loops
  - Block data transfer from program memory to data memory and from data memory to data memory
  - Bit or bit field manipulation of a register or I/O pins
  - Time delays creation
- **Key Computational Skills: Be able to**
  - Write loops to perform arithmetic and logic operation
  - Write macros
  - Manipulate bit

Electrical and Computer Engineering
Dalhousie University

IV. Hardware and Software Tools

- **Key Concept**
  - Understand the type of hardware development tools
  - Understand the type of software development tools
  - Use the AVR Studio IDE to enter, assemble, and download programs onto a board for execution
  - Use the AVR Studio IDE to debug AVR assembly programs
  - Use the AVR Studio IDE to debug AVR C programs
  - Learn the strategy for program debugging

V. Advanced Assembly Programming

- **Key Concept**
  - Issues related to subroutine calls
  - Stack pointers and operations
  - Subroutines to create time delays
  - Top-down-design methodology
  - Multiplication and division of signed and unsigned 16-bit or larger numbers
  - Write a subroutine to determine whether a number is a prime number.
- **Key Computational Skills: Be able to**
  - Read and write subroutines for arithmetic operation
  - Proper usage of stack pointers

VI. C Language Programming

- **Key Concept**
  - C structure, operator, basic data types and expressions
  - C loops, functions and function calls
  - Arrays and pointers for data manipulation
  - Basic I/O operations in the C
  - Bit and bit fields manipulation in C
  - Naming convention of AVR registers and bits
  - Use the AVR Studio IDE to enter, compile, and debug C programs (Lab)
- **Key Computational Skills: Be able to**
  - Read and write C function for arithmetic operation

VII. System Clock Configuration

- **Key Concept**
  - Understand the importance of the system clock generation
  - Understand the possible system clock generation option
  - Understand the Mega device clock circuit module
  - Be able to configure the Mega system clock signal by programming the fuse map

VIII. Parallel I/O

- **Key Concept**
  - Explain I/O synchronization methods
  - Verify voltage compatibility when interfacing a peripheral devices
  - Verify current compatibility when driving peripheral devices
  - Turn LED on and off
  - Input data from the DIP switches
  - Use seven-segment display to display numbers and letters
  - Use the I/O pin to generate waveform and make sound
  - Write programs to generate waveform using a D/A converter
- **Key Computational Skills: Be able to**
  - Configure the I/O and input and output data

IX. Advanced Parallel I/O

- **Key Concept**
  - Perform keypad and keyboard scanning, debouncing, and ASCII code lookup
  - Configure and display information on the LCD
  - Principles of operation of stepper motors
  - Write programs to drive stepper motors
  - Write programs to generate waveforms using a D/A converter
  - Write programs to perform data transfer using direct memory access (DMA)
- **Key Computational Skills: Be able to**
  - Write programs to drive stepper motors
  - Write programs to generate waveforms

X. Interrupt Handling, Resets, and Power Management

- **Key Concept**
  - Nature of interrupts and resets
  - How interrupts are handled by the CPU
  - Enable and disable interrupts
  - Set up interrupt priority
  - Write interrupt service routines
  - Use a watchdog timer reset to detect software errors
  - Use the low-power mode to save power consumption
- **Key Computational Skills: Be able to**
  - Enable and disable interrupts
  - Write interrupt service routines

XI. Timer Functions of the Mega AVR

- **Key Concept**
  - Overall structure of the Mega device timer system
  - Timer/counter operation
- **Key Computational Skills: Be able to**
  - Use the input-capture function to measure signal parameters, including frequency, pulse width, period, and duty cycle
  - Use the output-compare function to create time delays
  - Use the output-compare function to generate waveforms with a specified duty cycle
  - Use the timer force function to force the output-compare pin to either high or low
  - Configure the PWM function to generate periodic waveforms with a specified frequency and duty cycle

XII. USART

- **Key Concept**
  - Four aspects of the TIA-232 standard
  - Errors in data transmission and reception.
  - Null-modem connection
  - Operation of the USART subsystem
  - Wire the USART pins to the TIA-232 connector
- **Key Computational Skills: Be able to**
  - Configure the USART
  - Program the USART subsystem to perform data transmission and reception

XIII. The SPI Function

- **Key Concept**
  - SPI module of the Mega
  - MSPI mode of the USART
- **Key Computational Skills: Be able to**
  - Configure the SPI operation
  - Use SPI to interface with the shift register 74LV595
  - Use SPI to interface with the 12-bit D/A converter MCP4922
  - Use the SPI module to interface with the TC72 digital thermometer
  - Use MSPI mode of the USART to interface with the MC14489

XIV. Two-Wire Interface (TWI)

- **Key Concept**
  - I^2C and TWI protocol
  - I^2C and TWI signal components
  - I^2C and TWI bus arbitration methods
  - I^2C and TWI data transfer formats
- **Key Computational Skills: Be able to**
  - Configure TWI operation
  - Store and retrieve data in/from the serial EEPROM chip AT24C08B
  - Use the DS1631A to measure the ambient temperature
  - Use polling approach to transfer data via the TWI module

XV. Analog-to-Digital Converter

- **Key Concept**
  - A/D conversion process
  - The resolution, the various channels, and the operation modes of the Mega A/D converter
  - Interpret the A/D conversion results
  - Procedure for using the AVR Mega A/D converters
- **Key Computational Skills: Be able to**
  - Configure the A/D converter for the application
  - Use the humidity sensor IH-3610 from Honeywell
  - Use the barometric pressure sensor MP3H6115A from Freescale

Summary and Conclusion

- http://www.jasongu.org/3204
- Date: December 7 2017
- Time: 12:00 PM – 3:00 PM
- Location: Sexton Gym
- Closed-book exam with 4 page letter size formula sheets, both sides ok
- General Formula sheets will be provided and should be returned