Corwin Perren -931759527
ECE 152 – Thurs 10am
4/12/12
Assignment 1

- Write the binary output for the following equations:
  - 0b00111100 & 0b11110011
    - 0b00110000
  - 0b01010101 | 0b10101100
    - 0b11111101
  - (1<<3) | (1<<7)
    - 0b10001000
  - ~(0b10101111 | 0b11110000)
    - 0b00000000
  - 0b10000000 | (0b01111111 & 0b10110011)
    - 0b10111111
- List 1 similarity and 1 difference between application programs and embedded programs.
  - Similarity
    - They are both written by a programmer to perform a specific purpose
  - Difference
    - An application is meant to be able to run on many different types of hardware (PC's), while embedded programs are designed for and tied to their hardware.
- From our notes, what are the steps in the software design cycle? In your own words, describe each in complete sentences.
  - Specification
    - Specify what the software needs to accomplish and what is necessary to meet that goal.
  - Design
    - Preliminary prototypes of screen elements and functional elements, with the aim to find the best approach to meet requirements.
  - Coding
    - Write the actual code for the project, which changes based on the platform, be it application or embedded.
  - Testing
    - If a problem doesn't already exist, trying many different types of inputs to cause a problem, then fixing it and those that already exist.
- Explain what each line below does in detail.

	Description

void initialize (void) {	This is the header statement that creates a function called “initialize” with no or “void” arguments and a void return type. The bracket begins the area to hold the function’s code.
DDRA=0b00000000;	In setting the data direction register A to this binary value, the switches and buttons on the wunderboard are set as inputs to be read from.
DDRB=0b11000111;	Setting this register to this binary value sets the three of the sdcard reader pins to outputs, one of the sdcard pins to an input, the nothing and audio out pins to inputs (disabled), and the red and green color control pins to outputs.
DDRC=0b11111111;	This sets the led row control pins all to outputs.
DDRD=0b00000000;	This sets the serial tx and rx pins to inputs and otherwise have pins with nothing connected to them also set to inputs.
DDRE=0b00000111;	The first three bits set the column control pins for the led array to outputs, and otherwise sets a usb pin and relay pin to inputs (disabling them).
DDRF=0b00000000;	This sets every pin connected to an analog to digital converter to an input, including the accelerometer adc.
}	Closes the initialize function.
void InitializeUART (){	This makes function called “InitializeUART” with nothing as input arguments, and a void return type. The bracket begins the space for the function’s code.
UBRR1H = 0;	This register contains the four most significant bits for setting the USART baud rate and in this case is set to all zeroes.
UBRR1L = 6;	This is the other half of the previous register in which these bits for the baud rate are set to a six.
UCSR1C = 0b10100110;	From most significant to least significant, this sets serial clock polarity to the rising edge for transmission and falling edge for receiving, the number of serial data bits to 8, one serial stop bit, no parity for one register, even parity for the next, and the USART to a reserved mode through the last two bits.

UCSR1A = 0b00100000;	From most significant to least significant, this sets the wunderboard out of multi-processor communication mode, sets the serial connection to a synchronous connection, bits 2 through 5 are read only and tell the status of parity error checking and whether data registers are empty, and bit 6 and 7 show that the transmission and receiving have completed successfully.
UCSR1B = 1<<TXEN1 | 1<<RXEN1;	This sets the serial transmit and receive functionality to enabled and sets all other bits to zero.
}	Ends the InitializeUART function.