Adit Kotharii(Aditya Kothari)
ECE 375
LAB 1
Sources: AVR starter guide, microchip.com

Lab1: Study Questions

1.) Source code font: Courier New**,** Size**:** 8.

2.) Naming convention for source code file**:** Firstname_Lastname_Lab4_sourcecode.asm
Naming convention for source code file if working with a partner: Firstname_Last name_and_Firstname_Last name_Lab4_sourcecode.asm

3.) Pre-compiler Directives: are special instructions, denoted by a preceding dot. They are executed before the code is compiled.
Difference: .def: allows register to be defined by a symbol/label (example: .def test = r15), while, .equ allows symbol/label to have a value (.equ wtime = 100).

4.) (1 << 3) = (b00000001 << 3) = b00001000
(2 << 2) = (b00000010 << 2) = b00001000
(8 >> 1) = (b00001000 >> 1) = b00000100
(1 << 0) = (b00000001 << 0) = b00000001
(6 >> 1 | 1 << 6) = (b00000110 >> 1 | b00000001 << 6) = (b00000011 | b01000000) = b01000011

5.) **ADIW** = Add Immediate to word. It is an 8-bit arithmetic and logic instruction that uses ALU. It is responsible for addition.

BCLR = Bit clear in SREG. It is a Bit manipulation instruction, it allows the programmer to set and clear any bit within the SREG register.

BRCC = Branch if Carry Cleared.

BRGE = Branch if greater or equal signed.

COM = performs one's complement of register.

EOR = Exclusive OR. Performs XOR on contents of the register.

LSL = Logical shift left. It manipulates individual bits within 8 bit registers. Shifts all the bits in the register one place to the left, MSB to carry bit and shift in a 0 to LSB. (Multiplies by 2).

LSR = Logical shift right. It manipulates individual bits within 8 bit registers. Shifts all the bits in the register one place to the right, MSB is cleared and shift in LSB to carry. (divides by 2).

NEG = Two's complement. It is an 8-bit arithmetic and logic instruction that uses ALU. It changes the value of the register with its two's complement.

OR = Logical OR. Performs logical OR between contents of registers.

ORI = Logical OR with immediate. It is an 8-bit arithmetic and logic instruction that uses ALU. It performs logical OR between contents of register and a constant.

ROL = Rotate left through carry. It manipulates individual bits within 8 bit registers. It shifts all the bits in register one place to the left and shifts out the MSB to carry bit and shift in the carry bit to the LSB.

ROR = Rotate Right through carry. It manipulates individual bits within 8 bit registers. It shifts all the bits in register one place to the right and shifts out the LSB to carry bit and shift in the carry bit to the MSB.

SBC = Subtract with Carry. It is an 8-bit arithmetic and logic instruction that uses ALU. Subtracts two registers with the C flags.

SBIW = Subtract Immediate from Word. It is an 8-bit arithmetic and logic instruction that uses ALU. Subtracts an immediate value (0 to 63) from a register pair ad places the result in the register pair.

SUB = Subtract Without Carry**.** Subtracts two registers and places the value in the destination register.