ECE 375 PRELAB 5

Lab Time: Hao Truong 2-4

Hao Truong

QUESTIONS

1.For this lab, you will be asked to perform arithmetic operations on numbers that are larger than 8 bits. To be successful at this, you will need to understand and utilize many of the various arithmetic operations supported by the AVR 8-bit instruction set. List and describe all of the addition, subtraction, and multiplication instructions (i.e. ADC, SUBI, FMUL, etc.) available in AVR's 8-bit instruction set.

ADC: Adds two registers and the contents of the C Flag and places the result in the destination register Rd.

ADD: Adds two registers without the C Flag and places the result in the destination register Rd.

ADIW: Adds an immediate value (0 - 63) to a register pair and places the result in the register pair

SBC: Subtracts two registers and subtracts with the C Flag, and places the result in the destination register Rd.

SBCI: Subtracts a constant from a register and subtracts with the C Flag, and places the result in the destination register Rd.

SBIW: Subtracts an immediate value (0-63) from a register pair and places the result in the register pair.

SUB: Subtracts two registers and places the result in the destination register Rd.

SUBI: Subtracts a register and a constant, and places the result in the destination register Rd.

MUL: This instruction performs 8-bit × 8-bit → 16-bit unsigned multiplication.

MULS: This instruction performs 8-bit × 8-bit → 16-bit signed multiplication.

MULSU: This instruction performs 8-bit × 8-bit → 16-bit multiplication of a signed and an unsigned number.

2. Write pseudocode for an 8-bit AVR function that will take two 16-bit numbers (from data memory addresses $0111:$0110 and $0121:$0120), add them together, and then store the 16-bit result (in data memory addresses $0101:$0100). (Note: The syntax "$0111:$0110" is meant to specify that the function will expect little-endian data, where the highest byte of a multi-byte value is stored in the highest address of its range of addresses.)

$0100 <- $0110 + $0120 (add contents stored in low bytes and store in $0100)

$0101 <- $0111 + $0121 +carry (add contents stored in high bytes with carry and store to $0101)

3. Write pseudocode for an 8-bit AVR function that will take the 16-bit number in $0111:$0110, subtract it from the 16-bit number in $0121:$0120, and then store the 16-bit result into $0101:$0100.

$0100 <- $0120 - $0110 (subtract low byte and store in $0100)

$0101 <- $0121 - $0111 – carry (subtract with carry high byte and store in $0101)

REFERENCE

http://ww1.microchip.com/downloads/en/devicedoc/atmel-0856-avr-instruction-set-manual.pdf