Jose Franco Baquera

February 18, 2018

CS – 273

Homework #4

1. If you want to store the value 0x33A0 in register Y, what instructions will you need to do?

Register Y uses registers R29 and R28 as its parameters. Therefore, the instructions needed to store the value 0x33A0 into register Y are the following:

```
ldi r28, 0xA0

ldi r29, 0x33
```

(We note that the low byte is stored in r28 while the high byte is stored in r29.)

2. Why do you need to use the push and pop instructions to save register values in data memory?

We need to use the push and pop instructions to save register values in data memory because when other functions or subroutines are called, they may also use these same registers and may change and/or corrupt their original values. In other words, function calls may also use the same registers we are using and may corrupt/change the original values they had before the function call (i.e. the registers will have garbage data once the function call returns).

3. If a C function call to an assembly function passes four character type (char) variables with values 'm', 'o', 'o', 'n', which registers you need to retrieve to correctly receive the four characters in the assembly code?

If a C function call to an assembly function passes four characters type (char) variables (i.e. each character being one byte in size) with values 'm', 'o', 'o', 'n', then we would find each variable in the following registers:

- The variable with value 'm' would be saved in **R24**

- The variable with value of the **first** ‘o’ would be saved in **R22**
- The variable with value of the **second** ‘o’ would be saved in **R20**
- The variable with value ‘n’ would be saved in **R18**

In the assembly code, we would need to access from these four registers to correctly receive the four characters.

4. What is the difference between CPU and microcontroller? Is AVR a CPU or a microcontroller?

The difference between a CPU and a microcontroller is that a microcontroller has a processor, memory storage, and I/O capabilities all embedded onto a single chip while a CPU has external memory and I/O modules that are more separate and spread out. The AVR is a microcontroller.

5. How many bits are in the program counter register?

The program counter register has a total of 16 bits.

6. How do you read the 8 bits in the status register SREG to R18?

To read the 8 bits in the status register SREG to R18, we would use the following assembly instruction:

```
in R18, 0x3F
```

7. What are the operands of the two branching instructions?

The operand for the BRNE branching instruction is -3 (i.e. in signed binary, -3 is equal to 0b1111101) while the operand for the BREQ branching instruction is 1 (i.e. in signed binary, positive 1 is equal to 0b0000001). It is important to note that these binary numbers are signed.

8. What is the value of R20 at the return of this code?

The value of R20 (in decimal form) at the return of this code will be 16.