Danny Barnes
ECE 375 – R 10am-12pm
February 1, 2017

Lab 4 Prelab

1. The stack pointer is a register that holds the address of the top of the stack. The stack is used for storing temporary data such as local variables and return addresses from subroutine calls. We can initialize the stack pointer by writing data to the two 8-bit registers, Stack Pointer High Register and Stack Pointer Low Register. Here is pseudocode that initializes the stack pointer to the last SRAM address:

```
SPL ← LOW(RAMEND)
SPH ← HIGH(RAMEND)
```

2. The AVR instruction LPM, or Load Program Memory, loads one byte from program memory into a specified register or R0 if unspecified. The address for the data is retrieved from the Z register. To use LPM, an address must first be loaded into the Z register. Here is pseudocode that demonstrates LPM:

```
ZL ← LOW(ADDRESS)
ZH ← HIGH(ADDRESS)
LPM
```

3. The definition file `m128def.inc` contains symbolic keywords for many registers and locations within memory and IO, like the aforementioned SPL, RAMEND, and ZL. These can make the code much easier to write and interpret while also improving the ease with which it can be translated for use with a different device.