1. Objectives

In this lab, you will learn AVR programming on
- Stack, stack frame design and functions

2. Preparation

Before coming to the laboratory, you should:
2.1. obtain a memory stick on which to store your work;
2.2. read through this experiment in detail, trying to understand what you will be doing;
    and
2.3. write your programs at home in order to finish the experiment on time.

3. Task: String to Integer Conversion

The C program in Figure 1 implements the function of converting a string to an integer. The string is given in main() and its integer is obtained by calling function atoi(). Manually translate the program into an assembly program. Assume the string is stored in program memory and an integer takes two bytes.

```c
int main(void) {
  char s[] = "12345";
  int number;
  number = atoi(s);
  return 0;
}

int atoi(char *a) {
  char i;
  char c;
  int n;

  n = 0;
  c = *a;
  for (i=1;((c >='0') && (c <='9'));i++){
    n = 10 * n + (c -'0');
    c = *(a+i);
  }
  return n;
}

Figure 1: string_to_number.c
```
Make sure you use functions and stacks in your program. Be prepared to demonstrate the program using a stack diagram of the program, identifying stack frames and the places where the variables have to be stored.

4. Note:
The task is worth 5 marks. Your program should be well commented. Up to 1 mark will be deducted for the program without proper and sufficient comments.