

src\main.c

```
1
2 // Practice assignment 9, exercise 3
3
4 #include <stdio.h>
5 #include <avr/io.h>
6 #include <util/delay.h>
7
8 #include "usart.h" //init libs
9
10 char get_bit(char bit, char n) //same as previous exercise
11 {
12     if(n & (1 << bit))
13         return 1;
14     else return 0;
15 }
16
17 char clear_bit(char bit_number, char input_number) // function init. Returns numbers with bit
18 // cleared, done with an AND a reverse mask, made by NOT-ing 1 shifted by a number
19 {
20     return (input_number & ~(1 << bit_number));
21 }
22
23 char set_bit(char bit_number, char input_number) // Returns numbers with bit set with a mask and
24 // a OR
25 {
26     return (input_number |(1 << bit_number));
27 }
28
29 int main(void) {
30     char c, BitToSet, BitToClear, i; // initiate variables
31
32     uart_init(); // open the communication to the microcontroller
33     io_redirect(); // redirect input and output to the communication
34
35     while(1)
36     {
37         printf("Type in a number\n"); // message for user
38         scanf("%hhd", &c); //get number to mess with
39
40         for(i=7; i>=0; i--)
41             printf("%hhd ", get_bit(i, c)); // print byte before any changes, *backwards* because MSB is
42             // at the left of the number
43
44         printf("\nType in a bit to clear\n"); // message for user
45         scanf("%hhd", &BitToClear); //get bit to clear
```

```
46     c=clear_bit(BitToClear-1,c); //clear the bit by calling the function
47
48     for(i=7;i>=0;i--)
49         printf("%hhd ",get_bit(i,c)); //print byte afterwards
50
51     printf("\nType in a bit to set\n"); // message for user (has an endline at the beginning
because the previous line does not)
52     scanf("%hhd", &BitToSet); //get bit to set
53
54     c=set_bit(BitToSet-1,c); //print byte afterwards
55
56     for(i=7;i>=0;i--)
57         printf("%hhd ",get_bit(i,c)); //print byte afterwards
58
59     printf("\n"); // one final end of line for formatting reasons
60 }
61 }
62
```