

Coursework Assignment 1

Josh Murphy and Matthew Howard
Questions? [CS1 Stack Exchange](#)

Coursework

This is the first of two coursework assignments in CS1. It is worth 7.5% of the module. You will need to write an assembly program, which will then be submitted to KEATS. There is a base task that most students will be able to complete. The challenge task is more complex. Both tasks will be graded. You must work on the assignment individually.

Base Task: Have the Arduino display your k-number

You should modify the program you wrote in Lab 2 to write out the digits of your King's **K-number** to PORTB. You should reuse the circuit from Lab 2 as well. Name your new program `my_k_number.s`.

Have your program write out each digit of your K-number separately to PORTB, writing the left-most *numerical* digit first. For example, if your K-number is **K1070542**, then your program will first write out a **1**, followed by a **0**, then a **7**, then a **0**, then a **5**, then a **4**, and finally a **2**.

To help you understand how decimal digits correspond to their hexadecimal and binary representations, see the look-up table later in this document.

You will notice that as you start writing more than one number out to PORTB then only the *last* number that your program writes out is displayed on the LEDs. This is because your Arduino writes the numbers out *really, really, really fast* to the LEDs, and in fact, it's faster than the human eye can perceive.

In a future lab we will explore how we can add a *delay* between outputting to LEDs in order to perceive this change.

However, for now, test your program incrementally after adding code to display an additional digit from your k-number. That way, you can be sure that your program will output all of the digits for your K-number in the correct sequence.

Challenge Task: Printing your initials

You should now modify your program so that it also displays a binary encoding of your initials after it has finished displaying your K-number.

There are many ways to encode alphanumeric characters in binary, the most common is ASCII. However, we will use our own encoding of alphanumeric characters. We will assume an 'A' is the decimal value 1, a 'B' is 2, a 'C' is 3 and so on. In this encoding, 'Z' would be 26. Again, you can use the look-up table later in this document to find the equivalent binary values that you will display, and accompanying hexadecimal values.

You should also display a full stop character '.', which we will assume is encoded as the value 27, between your initials.

You will notice that for values greater than 15 there are not enough LEDs in the current circuit to display the correct value. You should modify the circuit appropriately so that it has the sufficient number of LEDs. You should give a brief description (no more than a few lines) of how you modified your circuit in a comment at the top of your program.

For example, Ada Lovelace's program would first display her K-number. The program would then display the value 1 (00001, representing 'a'), then the value 27 (11011, representing '.'), and then the value 12 (01100, representing 'l').

Decimal Digit	Hexademical Equivalent	Binary Number Representation							
0	0x00	0	0	0	0	0	0	0	0
1	0x01	0	0	0	0	0	0	0	1
2	0x02	0	0	0	0	0	0	1	0
3	0x03	0	0	0	0	0	0	1	1
4	0x04	0	0	0	0	0	1	0	0
5	0x05	0	0	0	0	0	1	0	1
6	0x06	0	0	0	0	0	1	1	0
7	0x07	0	0	0	0	0	1	1	1
8	0x08	0	0	0	0	1	0	0	0
9	0x09	0	0	0	0	1	0	0	1
10	0x0A	0	0	0	0	1	0	1	0
11	0x0B	0	0	0	0	1	0	1	1
12	0x0C	0	0	0	0	1	1	0	0
13	0x0D	0	0	0	0	1	1	0	1
14	0x0E	0	0	0	0	1	1	1	0
15	0x0F	0	0	0	0	1	1	1	1
16	0x10	0	0	0	1	0	0	0	0
17	0x11	0	0	0	1	0	0	0	1
18	0x12	0	0	0	1	0	0	1	0
19	0x13	0	0	0	1	0	0	1	1
20	0x14	0	0	0	1	0	1	0	0
21	0x15	0	0	0	1	0	1	0	1
22	0x16	0	0	0	1	0	1	1	0
23	0x17	0	0	0	1	0	1	1	1
24	0x18	0	0	0	1	1	0	0	0
25	0x19	0	0	0	1	1	0	0	1
26	0x1A	0	0	0	1	1	0	1	0
27	0x1B	0	0	0	1	1	0	1	1
		7	6	5	4	3	2	1	0

<bit position

Submission instructions

- You should submit a single .s file named `my_k_number.s` via KEATS.
- You must submit by ~~17/11/2019~~ 19/11/2019 by 5pm.
- Your submission will be marked for correctness, organisation, and readability.