

## Practical 7

All files submitted must be in the root of the zip folder. Running make on the contents of the zip should produce a single .elf file. When submitting, I suggest that you select ALL files in the working directory, zip them and submit. Do not submit a zipped folder as this causes errors in automarking.

For the following, only one switch will be interacted with at a time.

### Part 1: (2)

Please refer to the template file provided with two array definitions in it (patterns0[] and patterns1[]). The marker will change the values and length of these arrays at build time. Two pointers are also defined. The objective is to get these pointers to point to the two elements in either array that are the largest and smallest respectively. These pointers should point to the correct elements (from either array) by the time main() enters the while loop.

### Part 2: (2)

Please refer to the template file provided for an array outOfOrder[] defined. The marker will change the values and length of this array at build time. In whichever way you like, order the elements of the array from smallest to largest. Make sure that the array is ordered by the time main() enters the while loop.

### Part 3: (2)

Referring again to the template file provided with two array definitions in it. Every time SW0 is pressed (on the falling edge), display the next element in patterns0[] array. Begin from the first element and wrapping around. Every time SW1 is pressed (on the falling edge), display the next element in patterns1[] array. Begin from the first element and wrapping around. Holding the button should have no effect. This requires debouncing and will be tested for with noisy falling edges.

### Part 4: (2)

While SW2 is held down, POT0 to be sampled:  
If the pot is outputting 0 V, 0xFF should be displayed on the LEDs.  
If the pot is outputting 3.3 V, 0 should be displayed on the LEDs.  
Linear in between.  
Releasing SW2 should just leave the last sampled POT value on the LEDs.

**Total marks: 8**