

Lab 4: The C and V flags of an ARM MCU
Cameron Stark and Oleksandr Hendrik
Lab date: 2/13/19
Report Date: 2/13/19

INTRODUCTION

The purpose of the lab is to understand how the ARM processor understands and process the signed and unsigned number values, and how the programmer can deal with the results the ARM processor returns.

CODE

1. Explanation of the specific operations to generate $x[i][0]$ and $x[i][1]$

These numbers are generated by using the random number function, which is then modulo-ed with the maximum value of an unsigned integer and then added to the minimum value of a signed integer.

2. Explanation of the specific operations to generate the results and C and V Flags in Task 3

For the C flag, we check if the result of the subtraction is larger than x0 to determine the value of the C flag, 0 if true, and 1 if false.

For the V flag, we check for two specific conditions that produce an overflow the first being if negative - positive = positive and the second being positive - negative = negative, which is used to determine the value of the V flag, 0 if true and 1 if false.

3. The below screenshots represent Task 2 and Task 3 Results

The screenshot shows the ARM processor registers with the xPSR value 0x21000000. The flags are: N=0, Z=0, C=1, V=0, Q=0. The debug viewer shows the following output:

```
x[0] = 110, 94
110 - 94 = 16
result_u = 16, C flag = 1
0b0001_0000

110 - 94 = 16
result_i = 16, V flag = 0
0b0001_0000
```

The screenshot shows the ARM processor registers with the xPSR value 0x01000000. The flags are: N=0, Z=0, C=0, V=0. The debug viewer shows the following output:

```
x[1] = 56, -33
56 - 223 = 89
result_u = 89, C flag = 0
0b0101_1001

56 - -33 = 89
result_i = 89, V flag = 0
0b0101_1001
```


