

Introduction the DS4 and Functions

LAB 03

SECTION 6

Kenneth Schueman

SUBMISSION DATE:

9/23/2021

9/23/2021

Problem: DualShock 4 Data Collection

Move the DS4 around and notice how the values output by the program changes based on the controller orientation and direction. Collect some data samples.

Analysis

The step-by-step instructions made this a simple task

Design

No design necessary

Testing

Getting my values to look like the ones presented on the instructions took some testing.

Comments

Instructions were incomplete in order to achieve what was asked needed outside knowledge.

Screen Shots

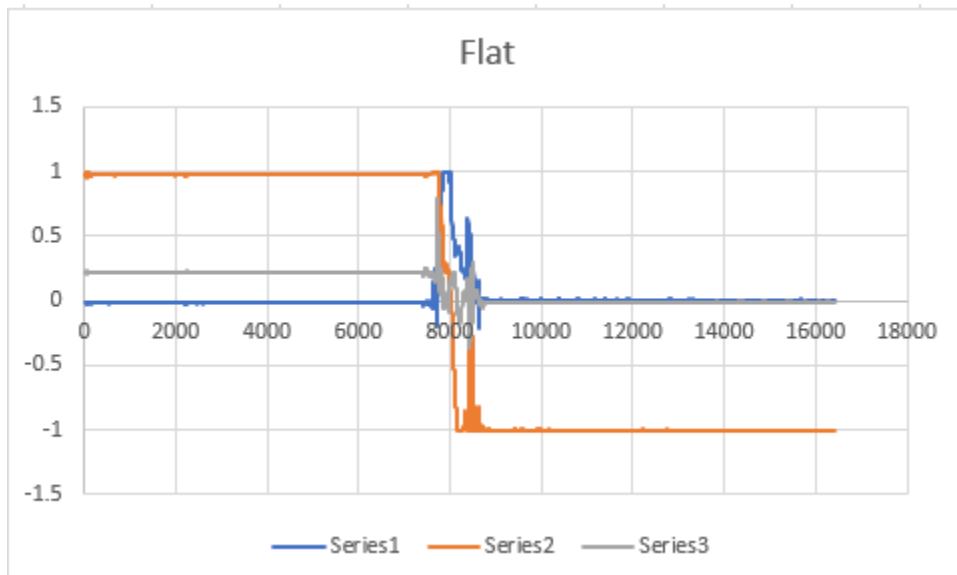


Figure 1

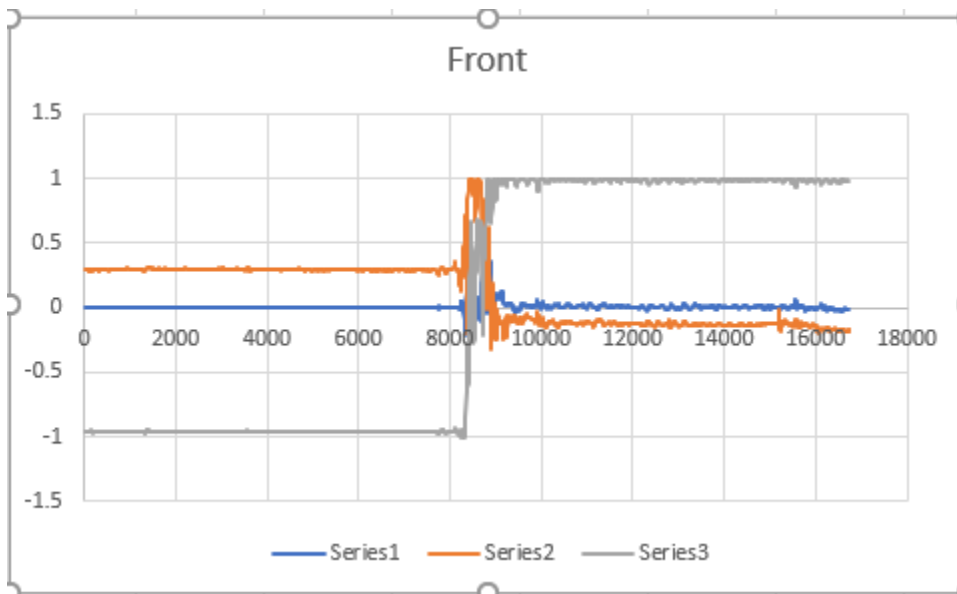


Figure 2

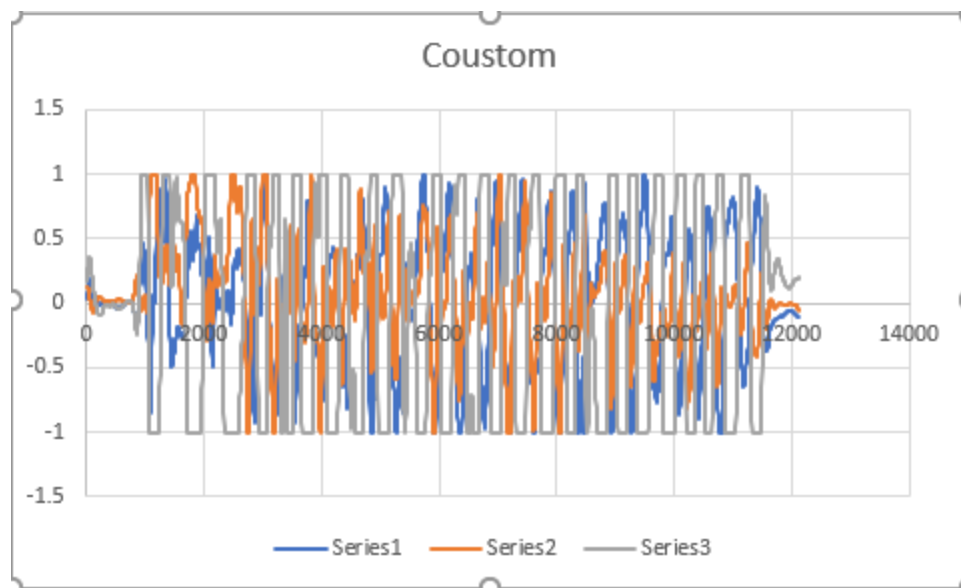


Figure 3

Problem: Introduction to Functions and the DualShock 4

Modify the line in SECTION 0 so that the milliseconds are printed out as SECONDS as a real number in an 8-character area with 3 decimal digits precision. Also modify the line so acceleration values are shown in a 7-character area with 4 digits of precision.

Analysis

This problem took way longer than it should have simply because I had a function dividing `t` by a 1000 and forgot to comment it out.

Design

Wrote three separate functions to handle the arithmetic of converted milliseconds

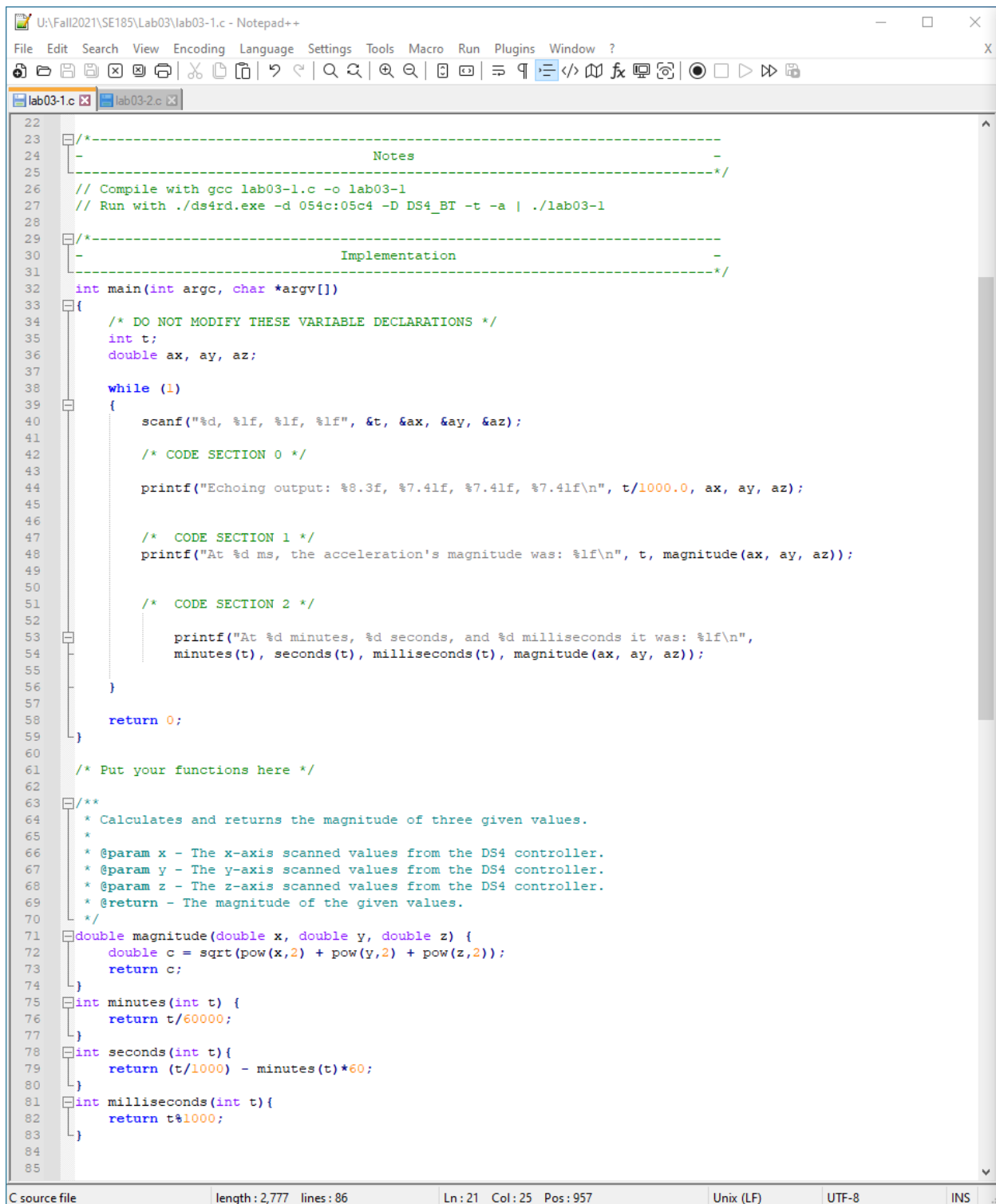
Testing

Once my math was correct there was a error were I was only getting 0 back because I had two separate inputs declared.

Comments

This was way harder then it should have been for me, but on the plus side I know understand functions very well.

Screen Shots



```
22
23  /*-----
24  -                               Notes                               -
25  -----*/
26  // Compile with gcc lab03-1.c -o lab03-1
27  // Run with ./ds4rd.exe -d 054c:05c4 -D DS4_BT -t -a | ./lab03-1
28
29  /*-----
30  -                               Implementation                       -
31  -----*/
32  int main(int argc, char *argv[])
33  {
34      /* DO NOT MODIFY THESE VARIABLE DECLARATIONS */
35      int t;
36      double ax, ay, az;
37
38      while (1)
39      {
40          scanf("%d, %lf, %lf, %lf", &t, &ax, &ay, &az);
41
42          /* CODE SECTION 0 */
43
44          printf("Echoing output: %8.3f, %7.4lf, %7.4lf, %7.4lf\n", t/1000.0, ax, ay, az);
45
46
47          /* CODE SECTION 1 */
48          printf("At %d ms, the acceleration's magnitude was: %lf\n", t, magnitude(ax, ay, az));
49
50
51          /* CODE SECTION 2 */
52
53          printf("At %d minutes, %d seconds, and %d milliseconds it was: %lf\n",
54                minutes(t), seconds(t), milliseconds(t), magnitude(ax, ay, az));
55
56      }
57
58      return 0;
59  }
60
61  /* Put your functions here */
62
63  /**
64   * Calculates and returns the magnitude of three given values.
65   *
66   * @param x - The x-axis scanned values from the DS4 controller.
67   * @param y - The y-axis scanned values from the DS4 controller.
68   * @param z - The z-axis scanned values from the DS4 controller.
69   * @return - The magnitude of the given values.
70   */
71  double magnitude(double x, double y, double z) {
72      double c = sqrt(pow(x,2) + pow(y,2) + pow(z,2));
73      return c;
74  }
75  int minutes(int t) {
76      return t/60000;
77  }
78  int seconds(int t){
79      return (t/1000) - minutes(t)*60;
80  }
81  int milliseconds(int t){
82      return t%1000;
83  }
84
85
```

C source file | length: 2,777 lines: 86 | Ln: 21 Col: 25 Pos: 957 | Unix (LF) | UTF-8 | INS

Figure 4

```
/cygdrive/U/Fall2021/SE185/Lab03
Echoing output: 180.660, -0.0162, 0.9683, 0.2178
At 180660 ms, the acceleration's magnitude was: 0.992582
At 3 minutes, 0 seconds, and 660 milliseconds it was: 0.992582
Echoing output: 180.664, -0.0171, 0.9856, 0.2236
At 180664 ms, the acceleration's magnitude was: 1.010793
At 3 minutes, 0 seconds, and 664 milliseconds it was: 1.010793
Echoing output: 180.668, -0.0225, 0.9819, 0.2174
At 180668 ms, the acceleration's magnitude was: 1.005965
At 3 minutes, 0 seconds, and 668 milliseconds it was: 1.005965
Echoing output: 180.672, -0.0166, 0.9696, 0.2228
At 180672 ms, the acceleration's magnitude was: 0.995006
At 3 minutes, 0 seconds, and 672 milliseconds it was: 0.995006
Echoing output: 180.676, -0.0153, 0.9749, 0.2133
At 180676 ms, the acceleration's magnitude was: 0.998024
At 3 minutes, 0 seconds, and 676 milliseconds it was: 0.998024
Echoing output: 180.680, -0.0172, 0.9888, 0.2249
At 180680 ms, the acceleration's magnitude was: 1.014161
At 3 minutes, 0 seconds, and 680 milliseconds it was: 1.014161
Echoing output: 180.684, -0.0186, 0.9769, 0.2195
At 180684 ms, the acceleration's magnitude was: 1.001452
At 3 minutes, 0 seconds, and 684 milliseconds it was: 1.001452
Echoing output: 180.688, -0.0161, 0.9659, 0.2146
At 180688 ms, the acceleration's magnitude was: 0.989625
At 3 minutes, 0 seconds, and 688 milliseconds it was: 0.989625
Echoing output: 180.692, -0.0144, 0.9825, 0.2261
At 180692 ms, the acceleration's magnitude was: 1.008320
At 3 minutes, 0 seconds, and 692 milliseconds it was: 1.008320
Echoing output: 180.696, -0.0187, 0.9745, 0.2186
At 180696 ms, the acceleration's magnitude was: 0.998885
At 3 minutes, 0 seconds, and 696 milliseconds it was: 0.998885
Echoing output: 180.700, -0.0194, 0.9766, 0.2180
At 180700 ms, the acceleration's magnitude was: 1.000791
At 3 minutes, 0 seconds, and 700 milliseconds it was: 1.000791
Echoing output: 180.704, -0.0142, 0.9819, 0.2263
At 180704 ms, the acceleration's magnitude was: 1.007777
At 3 minutes, 0 seconds, and 704 milliseconds it was: 1.007777
Echoing output: 180.708, -0.0179, 0.9714, 0.2151
At 180708 ms, the acceleration's magnitude was: 0.995124
At 3 minutes, 0 seconds, and 708 milliseconds it was: 0.995124
Echoing output: 180.712, -0.0172, 0.9800, 0.2184
At 180712 ms, the acceleration's magnitude was: 1.004166
At 3 minutes, 0 seconds, and 712 milliseconds it was: 1.004166
Echoing output: 180.716, -0.0173, 0.9801, 0.2218
At 180716 ms, the acceleration's magnitude was: 1.005037
At 3 minutes, 0 seconds, and 716 milliseconds it was: 1.005037
Echoing output: 180.720, -0.0188, 0.9751, 0.2169
At 180720 ms, the acceleration's magnitude was: 0.999111
At 3 minutes, 0 seconds, and 720 milliseconds it was: 0.999111
Echoing output: 180.724, -0.0176, 0.9788, 0.2185
At 180724 ms, the acceleration's magnitude was: 1.003008
At 3 minutes, 0 seconds, and 724 milliseconds it was: 1.003008
Echoing output: 180.728, -0.0165, 0.9761, 0.2183
At 180728 ms, the acceleration's magnitude was: 1.000315
At 3 minutes, 0 seconds, and 728 milliseconds it was: 1.000315
Echoing output: 180.732, -0.0190, 0.9752, 0.2214
At 180732 ms, the acceleration's magnitude was: 1.000225
At 3 minutes, 0 seconds, and 732 milliseconds it was: 1.000225
Echoing output: 180.736, -0.0193, 0.9800, 0.2219
At 180736 ms, the acceleration's magnitude was: 1.004979
At 3 minutes, 0 seconds, and 736 milliseconds it was: 1.004979
Echoing output: 180.740, -0.0176, 0.9752, 0.2196
At 180740 ms, the acceleration's magnitude was: 0.999794
At 3 minutes, 0 seconds, and 740 milliseconds it was: 0.999794
Echoing output: 180.744, -0.0181, 0.9756, 0.2211
At 180744 ms, the acceleration's magnitude was: 1.000483
At 3 minutes, 0 seconds, and 744 milliseconds it was: 1.000483
Echoing output: 180.748, -0.0190, 0.9766, 0.2211
At 180748 ms, the acceleration's magnitude was: 1.001453
At 3 minutes, 0 seconds, and 748 milliseconds it was: 1.001453
Echoing output: 180.752, -0.0165, 0.9792, 0.2229
At 180752 ms, the acceleration's magnitude was: 1.004431
At 3 minutes, 0 seconds, and 752 milliseconds it was: 1.004431
```

Figure 5

Problem: Counting Buttons

Never was able to achieve a working program 😞