

```

1  /*****
2  * Senior Design Project: Electronic "Soft Differential"
3  * Credit: UNCA-NCSU Senior Class 2014
4  * Purpose: The purpose of this program is to read in various inputs from two wheel speed
5  *           encoders, one steering potentiometer, and one throttle potentiometer. The
6  *           code will determine if the car is straight or turning, and by how much it is
7  *           turning. An algorithm will process this and send outputs to two electric
8  *           motors which will run at different speeds in a turn to simulate what is
9  *           commonly seen in a regular mechanical differential.
10 *
11 * Versions & Version Notes:
12 *   v0.1.0- Code written for analog input of potentiometer inputs for steering and throttle
13 *           Credit: Dakota Lazenby
14 *   v0.1.1- Added support for motor encoders and modified structure
15 *           Credit: Dakota Lazenby
16 *   v1.0.0- Ported code over for Arduino Due and added throttle response functionality
17 *           Credit: Dakota Lazenby
18 *   v1.0.1- Imported simplified working motor encoder code
19 *           Credit: Brandon Zschokke, Jennifer Cory, and Hallie Sheaffer
20 *   v2.0.0- Combined and created confirmed working throttle code and encoder code
21 *           Credit: Brandon Zschokke
22 *   v2.1.0- Re-Located the encoder code to a function, cleaned up some variables and added conditional prints
23 *           Credit: Dakota Lazenby
24 *   v2.2.0- Added in a throttleFilter function.
25 *           Credit: Dakota Lazenby
26 *   v2.3.0- Added functionality for power LED, checking pack voltage from LiPo battery, new encoder Z pin reads
27 *           at high RPMs,
28 *           Credit: Dakota Lazenby, Hallie Shaeffer
29 *
30 * Future Versions / Roadmap:
31 *   v2.x.x- Implement Hallie's SD Card code for data logging. Implement Steering Code
32 *   v3.0.0- Implement controller code
33 *   v3.x.x- Debug and refine control code
34 *   v4.0.0- Remove excess global variables and modularize code into various functions
35 *****/
36
37 #define DEBUG      1
38 #define USER_INPUT 1
39 #define ENCODERS    1
40 #define SPEED       1
41 #define ADC_DELAY 10
42 // Global Declaration Section
43
44 int controlScheme = 1;
45 int loopCount = 0;
46
47 // Throttle Declarations
48 int throttlePin = A11; // Throttle Pot Signal
49 int leftMotor = DAC0;  // To Protoboard (5)
50 int rightMotor = DAC1; // To Protoboard (6)
51 int minThrottle = 7;
52 int maxThrottle = 235; // This value needs to correspond to the maximum voltage reading off of the potentiometer
53                       // plugged into the formula : maxThrottle = measuredVoltage * (3.3/(2^Res))
54 int throttle_in_left = 0;
55 int throttle_in_right = 0;
56
57 int Res = 10; // Resolution
58 int noiseDelay = 10; // Delay to ensure that analogReads are performed correctly
59

```