

## **Electronics Changelog**

### **Version 1.0**

- 16Mhz Arduino NANO
  - 3 Hardware Timers (Timer0, Timer1 and Timer2)
  - 1 UART (serial port)
  - 6 PWM Pins (8-bit)
  - 2 Interrupt Pins (Pin 2 & Pin 3)
  - 2 Kb RAM
  - 32 Kb Flash ROM
- (2) BTS7960 43A Motor Drivers
- Input from PS2 Style Joystick
- Hall Effect Sensors for wheel encoders (speed sensor)

### **Limitations of Version 1.0 Hardware**

After building the 1.0 electronics test platform, we quickly determined there were some limitations of using such a slow 8-bit processor. First, because each of the BTS7960 motor drivers required 2 PWM signals, we needed a total of 4 PWM outputs. Using 4 PWM pins on the Arduino Nano meant that both TIMER0 and TIMER1 were being used to generate PWM signals and we only had TIMER2 left. We are using Timer2 set to 500ms to average the wheel encoder counts. Also, we have two HALL effect sensors we are using to measure wheel speed, these use up both Interrupt pins, leaving us with no free external interrupt pins in case we want to measure something else using interrupts. Two PID controllers manage motor speed based upon user input and measured speed.

### **Version 2.0:**

- 16Mhz Arduino NANO
  - 3 Hardware Timers (Timer0, Timer1 and Timer2)
  - 1 UART (serial port)
  - 6 PWM Pins (8-bit)
  - 2 Interrupt Pins (Pin 2 & Pin 3)
  - 2 Kb RAM
  - 32 Kb Flash ROM
- (1) Cytron SmartDuo 60A Dual Channel Motor Driver (supports UART)
- Input from Accelerator Pedal + Steering Wheel
- (2) Temperature Sensors
- Hall Effect Sensors for wheel encoders (speed sensor)

### **Notes on Version 2.0 Hardware**

The biggest change to this revision is the change from the generic motor drivers in V1 to the Cytron 60A Dual Driver in V2. This new motor driver allows serial communication (UART), allowing us to free up another TIMER in the Arduino, by removing the need for PWM, which will allow us to smoothly apply acceleration and deceleration, preventing the jerky movements we experienced in Version 1.0.