Electronic Training and Activities

1. 01\_IMG\_2614.jpg
   * Robotic Car from the Georgia Tech Mechatronics training class.
   * Uses the TI MSP432 microcontroller and implements 2 motors with PWM control, numerous bump sensors, proximity sensor, and UART serial communication to the laptop.
2. 02\_IMG\_2748.MOV and 02\_IMG\_2749.MOV
   * Load Sensor Lab from the Univ of Colorado Embedding Sensors and Motors Specialization.
   * Uses the Cypress PSOC 5LP microcontroller, load cell, uA741 amplifiers, LCD screen with parallel communication, and PicoScope oscilloscope.
3. 03\_IMG\_2898.jpg
   * PID Lab from the Univ of Colorado Embedding Sensors and Motors Specialization.
   * Uses the Cypress PSOC 5LP microcontroller and PicoScope oscilloscope.
   * Programmed a PID (Proportional, Integral, Derivative) Control Loop to control a motor to reach the desired output.
4. 04\_IMG\_3171.MOV
   * Independent experimentation with the 555 timer IC setup in astable mode to produce a PWM signal to drive a DC electric motor.
   * A potentiometer is used to adjust the resistance in the circuit and therefore the speed of the motor.
5. 05\_IMG\_3168.MOV
   * Independent experimentation with the 555 timer IC setup in monostable mode to run the motor at the push of a button and to continue running for a specific time period.
   * The size of the capacitors in the system determine the length of time the motor is run.
6. 06\_IMG\_3170.MOV
   * Independent experimentation with the 555 timer IC setup in bistable mode to run the motor at the push of a button and to stop the motor at the push of a second button.
7. 07\_IMG\_3172.MOV
   * Independent experimentation with the Cypress PSOC 5LP microcontroller and the 555 timer IC setup in astable mode.
   * The microcontroller is programmed to start and stop the motor for specific time periods.
8. 08\_IMG\_3181.MOV
   * Independent experimentation with the Cypress PSOC 5LP microcontroller.
   * The microcontroller is programmed to put out a PWM signal to drive the motor. The PWM signal is stepped up after specific periods of time to increase the speed of the motor. The PWM signal is then gradually increased from off to full speed and then gradually decreased back to off.
9. 09\_IMG\_3276.MOV
   * Independent experimentation with the Cypress PSOC 5LP microcontroller and a L293D H-Bridge motor driver.
   * The microcontroller is programmed to put out a PWM signal to drive the motor. The motor driver is used to allow the motor to be spun in both the forward and reverse directions.
10. 10\_IMG\_3332.MOV
    * Independent experimentation with the Cypress PSOC 5LP microcontroller and a HM-18 Bluetooth Module.
    * The HM-18 Bluetooth Module allows for serial communication with an iphone. A basic joystick app was used to send different signals to the microcontroller. The microcontroller was programmed to interpret those signals and set the motor to 4 different speeds or off depending on which button was pushed.
11. 11\_IMG\_3339.MOV
    * Independent experimentation with the Cypress PSOC 5LP microcontroller, HM-18 Bluetooth Module, and a DRV8833 motor driver.
    * The HM-18 Bluetooth Module allows for serial communication with an iphone. A basic joystick app was used to send different signals to the microcontroller. The microcontroller was programmed to interpret those signals and set the motor to 2 different motor speeds in both forward and reverse.
12. 12\_IMG\_3355.MOV
    * Independent experimentation with the Cypress PSOC 5LP microcontroller, 28BYJ-48 Stepper Motor, ULN2003 Stepper Motor Driver Board.
    * The stepper motor was controlled with a custom program to move back and forth between set positions. The program was written to take individual steps using Unipolar mode of the stepper motor. The method of movement is by taking half-steps.
13. 13\_IMG\_3370.MOV
    * Independent experimentation with the Cypress PSOC 5LP microcontroller, 28BYJ-48 Stepper Motor, ULN2003 Stepper Motor Driver Board and HM-18 Bluetooth Module.
    * The stepper motor is controlled through a terminal app on a cell phone which sends text inputs to the microcontroller via UART communication using Bluetooth. A custom program was written to control the stepper motor with the following commands: ‘+’ will take individual steps forward, ‘-‘ will step backwards, ‘S’ will define a new home position, and ‘H’ will reverse the stepper motor back to its home position.