**Comparison**

**Codes:**

* MPU6050\_2DOF\_FEEDBACK\_Alt\_I2C\_DMP\_PID.ino & PendPID.ino
  + No much changes on these two files
  + Some tests for sensors are crossed out
  + No print or export for part of data
* MPU6050\_2DOF\_FEEDBACK\_Alt\_I2C\_DMP\_PD.ino & PID2DOFPend.ino
  + Adding a new board Arduino mini pro on the pendulum, the basic test functions and debuggers were added based on Charlie’s codes
  + May have something wrong about processing the angle data from the DMP sensor fusion from Charlie. This part was marked in Jee’s codes.
  + No big change on the Arduino Mega
* DriveCart.ino & PID2DOFPend.ino
  + PID2DOFPend.ino was written not based on DriveCart.ino
  + Can balance the pendulum when considering the cart as a disturbance
  + The entire thing should be rewritten if want to design a controller that balance the cart while keep moving for a long time
  + Shorten the bytes of data that the program can run faster and saving the storage space
* MPU6050\_calibration.ino
  + From the open source; some problems of Charlie’s codes are fixed by Jee.
  + "Digital Motion Processor" (DMP), also called a "Digital Motion Processing Unit", which can be programmed with firmware and is able to do complex calculations with the sensor values.  
    The DMP ("Digital Motion Processor") can do fast calculations directly on the chip. This reduces the load for the microcontroller (like the Arduino). The DMP is even able to do calculations with the sensor values of another chip, for example a magnetometer connected to the second (sub)-I2C-bus.
* TestSketches folder is used to test various components
  + TestDC
  + TestMPU
  + TestRotor
  + TestSerialCart
  + TestSerialPend
  + TestThrust

**Problems:**

* + The entire codes about balance the cart should be rewritten if want to design a controller that balance the cart while keep moving for a long time. A better coding skill is required.