```
#ifndef _CHASSIS_H_
#define _CHASSIS_H_
// Place all chassis related globals, function stub declartions in this
// header file.
//

#define WHEEL_DIAMETER 4      // diameter of drive wheels
#define WHEEL_BASE 18         // wheel base in inches left to right wheel diagonal measured
                              // of chassis.

#define DRIVE_MODE 2           // 1 for x-drive TANK
                               // 2 for x-drive ARDCADE
                               // 3 for standard TANK
                               // 4 for standard ARCADE
                               // 5 for Ryan Mode (arcade but with the arrow keys for movement)

#define JOY_SCALE .5           // scale the joysticks to 50% valid values are 0 - 1
                               // note: 0 will disable joysticks 0 * x = 0 always....

#define DEAD_STICK 10          // Dead stick zone - joystick movement between
                               // -DEAD_STICK and DEAD_STICK will result in a joystick
                               // value of 0 - dealing with slow creepign of robot when joysticks
                               // do not land perfectly in the center position

#define MOTOR_MAX_SPEED 200    // Robot has 36 gearset (RED) for 18 gearset (GREEN) -- 200
                               // for 06 (BLUE) -- 600

void driveRobot(int speed);    // function declaration - drives robot forward/backward
                               // indefinite untill next action

// Sets the speeds of the left and right wheels of the chassis
void chassisSetOpcontrol(int left, int right);

// Sets the speeds of each motor individually.
// The order of arguments is Front Right, Front Left, Back Right, Back Left
void setIndividualMotor(int FRight, int FLeft, int BRight, int BLeft);

// Create a convienant function to stop drive motors
void chassisStopDrive();

// left turn function - robot will spin (pivot turn) indifinete
void turnLeft(int speed);

// right turn function - robot will spin (pivot turn) indefinite
void turnRight(int speed);

// drive with PID control for a given distnace in inches
void driveForDistancePID(int distance, int speed);

// make a pivot turn to the right or left for a given angle - angle
//should be greater the 10 and less equal to 360
void pivotTurn(int speed, long angle);

int average(int x, int y);

#endif // _CHASSIS_H_

```