

```

#include "vex.h"
#include "math.h"
#define PI 3.14159265
using namespace vex;
////////////////////////////////////

//Odometry Task////////////////////////////////////

//Initial values
double initial_Dir;
double initial_x;
double initial_y;

//X and Y coordinates
double x;
double y;

//Current values
double Dir;
double l_dist;
double r_dist;
double b_dist;

//Previous values
double prev_Dir;
double prev_L;
double prev_R;
double prev_B;

//Combo values
double Dir_diff = Dir - prev_Dir;
double r_diff = r_dist - prev_R;
double l_diff = l_dist - prev_L;
double b_diff = b_dist - prev_B;

double rR;          //translation value
double bR;

double l_dfc = 5;   //left track wheel distance from center
double r_dfc = 5;   //right track wheel distance from center
double b_dfc = 5;   //back track wheel dist from center
double twc = 10.2102; //track wheel circumference - (3.25 * 3.14159)

int OdometryTask()
{
    while(1)
    {
        l_dist = LTrack.position(turns) * twc;
        r_dist = RTrack.position(turns) * twc;
        b_dist = BTrack.position(turns) * twc;
        Dir = (((l_dist - r_dist) / (l_dfc + r_dfc)) * (180 / 3.14159)) + initial_Dir;

        rR = (r_diff / Dir_diff) + r_dfc;
        bR = (b_diff / Dir_diff) + b_dfc;
    }
}

```

```

x = (2 * sin(Dir / 2)) * bR;
y = (2 * sin(Dir / 2)) * rR;

Brain.Screen.printAt(175, 60, "Gyro %.3f", Dir); //Output to brain screen
Brain.Screen.printAt(175, 120, "x %.1f", x);
Brain.Screen.printAt(175, 180, "y %.1f", y);
Controller1.Screen.setCursor(3, 1);
Controller1.Screen.print("Gyro %.2f", Dir);
printf("Gyro %.4f", Dir);
printf(" X_Pos %.2f", x);
printf(" Y_Pos %.2f\n", y);

vex::task::sleep(15);

prev_Dir = Dir;
prev_L = l_dist;
prev_R = r_dist;
prev_B = b_dist;
}
return(0);
}

vex::task o(OdometryTask); //Start Odometry Task

```