//non robot includes

#include <iostream>

#include <memory>

#include <string>

//robot related includes

#include "WPILib.h"

#include <IterativeRobot.h>

#include <LiveWindow/LiveWindow.h>

#include <SmartDashboard/SendableChooser.h>

#include <SmartDashboard/SmartDashboard.h>

#include <PowerDistributionPanel.h>

#include <CANTalon.h>

class Robot: public frc::IterativeRobot {

private:

//std::shared\_ptr<NetworkTable> roboRealm;

public:

//Drive train right side

CANTalon \*DTR1, \*DTR2, \*DTR3;

//Drive train left side

CANTalon \*DTL1, \*DTL2, \*DTL3;

//S is shooter I is intake

CANTalon \*S1, \*S2, \*I;

//Conveyer

CANTalon \*C1, \*C2;

//Driver controller

Joystick \*driver;

//Operator controller

Joystick \*op;

PowerDistributionPanel \*pdp;

Robot()

{

//Sets the values in the daisy chain of the talons for the right side

DTR1 = new CANTalon(1);

DTR2 = new CANTalon(2);

DTR3 = new CANTalon(3);

//Sets the values in the daisy chain of the talons for the left side

DTL1 = new CANTalon(10);

DTL2 = new CANTalon(11);

DTL3 = new CANTalon(12);

//Sets the values in daisy chain of the talons in the shooter/intake

S1 = new CANTalon(4);

S2 = new CANTalon(5);

I = new CANTalon(6);

//Sets the values in the daisy chain of the talons in the conveyer

C1 = new CANTalon(7);

C2 = new CANTalon(8);

//Sets the port for the Driver controller to USB port 0

driver = new Joystick(0);

//Sets the port for the operator controller to USB port 1

op = new Joystick(1);

pdp = new PowerDistributionPanel();

}

void TeleopPeriodic()

{

double t1 = pdp->GetCurrent(15);

double t2 = pdp->GetCurrent(14);

double t3 = pdp->GetCurrent(13);

double t4 = pdp->GetCurrent(12);

double t5 = pdp->GetCurrent(11);

double t6 = pdp->GetCurrent(10);

double t7 = pdp->GetCurrent(8);

double t8 = pdp->GetCurrent(7);

double t9 = pdp->GetCurrent(0);

double t10 = pdp->GetCurrent(1);

double t11 = pdp->GetCurrent(2);

//Sends the values of the y axis's of the tankDrive function

tankDrive(driver->GetRawAxis(1),driver->GetRawAxis(5));

//printf("current0: %f, current1: %f, current2: %f, current3: %f, current4: %f, current5: %f", current0, current1, current2, current3, current4, current5);

//printf("current6: %f, current7: %f, current8: %f, current9: %f, current10: %f, current11: %f", current6, current7, current8, current9, current10, current11);

//A button to control shooter and intake

if(driver->GetRawButton(6))

{

printf("t1: %f, t2: %f, t3: %f, t4: %f, t5: %f, t6: %f \n", t1, t2, t3, t4, t5, t6);

//printf("t7: %f, t8: %f, t9: %f, t10: %f, t11: %f \n", t7, t8, t9, t10, t11);

}

/\*if(op->GetRawButton(0))

{

S1->Set(.5);

S2->Set(.5);

I->Set(.5);

}

//X button to control convayer

if(op->GetRawButton(2))

{

C1->Set(.5);

C2->Set(.5);

}

//B button to reverse convayer

else if(op->GetRawButton(1))

{

C1->Set(-.5);

C2->Set(-.5);

}\*/

}

void tankDrive(double left, double right)

{

//Assigns left side of the drive train to the y axis of the left joy stick of the X-box controller

//DTL1->Set(-left);

DTL2->Set(-left);

DTL3->Set(-left);

//Assigns right side of the drive train to the y axis of the right joy stick of the X-box controller

DTR1->Set(right);

DTR2->Set(right);

DTR3->Set(right);

}

void RobotInit()

{

}

};

START\_ROBOT\_CLASS(Robot)