package edu.wpi.first.wpilibj.templates;

import edu.wpi.first.wpilibj.Compressor;

import edu.wpi.first.wpilibj.IterativeRobot;

import edu.wpi.first.wpilibj.Joystick;

import edu.wpi.first.wpilibj.Relay;

import edu.wpi.first.wpilibj.RobotDrive;

import edu.wpi.first.wpilibj.Timer;

import edu.wpi.first.wpilibj.Victor;

import edu.wpi.first.wpilibj.smartdashboard.SmartDashboard;

public class RobotTemplate extends IterativeRobot {

RobotDrive robotDrive = new RobotDrive(10, 2, 4, 3);

Joystick leftJoystick = new Joystick(1);

Joystick rightJoystick = new Joystick(2);

Joystick controller = new Joystick(3);

Victor shooter1 = new Victor(6);

Victor shooter2 = new Victor(7);

Compressor compressor = new Compressor(1,1);

Relay hang = new Relay(3);

Relay discIn = new Relay(4);

Relay shooterAngle = new Relay(5);

int currentShots;

int desiredShots;

boolean prevHang;

boolean prevPush;

public void robotInit() {

compressor.start(); //Starts Compressor

}

public void autonomousInit() {

robotDrive.setSafetyEnabled(false);

discIn.set(Relay.Value.kReverse);

shooter1.set(1);

shooter2.set(1);

currentShots = 0;

desiredShots = 3;

Timer.delay(3.0);

}

public void autonomousPeriodic() {

if (currentShots < desiredShots) {

discIn.set(Relay.Value.kReverse);

Timer.delay(1.5);

discIn.set(Relay.Value.kForward);

Timer.delay(1.5);

currentShots++;

} else {

Timer.delay(1.0);

shooter1.set(0.0);

shooter2.set(0.0);

discIn.set(Relay.Value.kReverse);

}

}

public void teleopInit() {

prevHang = false;

prevPush = false;

robotDrive.setSafetyEnabled(true);

}

public void teleopPeriodic() {

if (controller.getAxis(Joystick.AxisType.kY) > 0.25) {

shooterAngle.set(Relay.Value.kForward);

}

else if (controller.getAxis(Joystick.AxisType.kY) < -0.25) {

shooterAngle.set(Relay.Value.kReverse);

} else {

shooterAngle.set(Relay.Value.kOff);

}

if (controller.getRawButton(2)) {

if (!prevHang) {

hang.set(Relay.Value.kForward);

} else {

hang.set(Relay.Value.kOff);

}

prevHang = true;

} else {

if (prevHang) {

hang.set(Relay.Value.kReverse);

} else {

hang.set(Relay.Value.kOff);

}

prevHang = false;

}

if (controller.getRawButton(4)) {

shooter1.set(1);

shooter2.set(1);

} else {

shooter1.set(0);

shooter2.set(0);

}

if (controller.getRawButton(6)) {

if (!prevPush) {

discIn.set(Relay.Value.kForward);

} else {

discIn.set(Relay.Value.kOff);

}

prevPush = true;

} else {

if (prevPush) {

hang.set(Relay.Value.kReverse);

} else {

hang.set(Relay.Value.kOff);

}

prevPush = false;

}

robotDrive.tankDrive(leftJoystick.getAxis(Joystick.AxisType.kY), rightJoystick.getAxis(Joystick.AxisType.kY), false);

SmartDashboard.putNumber("Left Wheels", leftJoystick.getAxis(Joystick.AxisType.kY));

SmartDashboard.putNumber("Right Wheels", rightJoystick.getAxis(Joystick.AxisType.kY));

SmartDashboard.putNumber("Shooter", (shooter1.get() + shooter2.get()) / 2);

SmartDashboard.putNumber("Shooter Angle", controller.getAxis(Joystick.AxisType.kY));

SmartDashboard.putBoolean("Pushing", prevPush);

SmartDashboard.putBoolean("Hanging", prevHang);

}

}