#include "WPILib.h"

class Robot: public IterativeRobot

{

public:

Joystick \*leftJoy, \*rightJoy, \*controller;

CANTalon \*driveTrainR1, \*driveTrainR2, \*driveTrainR3, \*driveTrainL1, \*driveTrainL2, \*driveTrainL3;

CANTalon \*RArm1, \*RArm2, \*LArm1, \*LArm2,\*armExt1, \*armExt2;

int initalEncT1;

int initalEncT3;

int T1Enc;

int T2Enc;

int auton = 0;

int autonMax = 2;

Robot()

{

//x-box

controller = new Joystick(0);

//sticks

rightJoy = new Joystick(1);

leftJoy = new Joystick(2);

//Right Drive Train

driveTrainR1 = new CANTalon(0);

driveTrainR2 = new CANTalon(1);

driveTrainR3 = new CANTalon(2);

//Left Drive Train

driveTrainL1 = new CANTalon(3);

driveTrainL2 = new CANTalon(4);

driveTrainL3 = new CANTalon(5);

//Right Arm

RArm1 = new CANTalon(6);

RArm2 = new CANTalon(7);

//Left Arm

LArm1 = new CANTalon(8);

LArm2 = new CANTalon(9);

//Arm Extenders

armExt1 = new CANTalon(10);

armExt2 = new CANTalon(11);

//Slaving left arm to right arm

LArm1->SetControlMode(CANSpeedController::kFollower);

LArm2->SetControlMode(CANSpeedController::kFollower);

LArm1->Set(6);

LArm2->Set(6);

//Slaving arm extenders

armExt1->SetControlMode(CANSpeedController::kFollower);

armExt1->Set(10);

}

void RobotInit()

{

}

void DisabledPeriodic()

{

SmartDashboard::PutString("DB/String 7","");

if(controller->GetRawButton(6))

{

auton++;

Wait(.2);

if(auton == autonMax + 1)

{

auton = 0;

}

else if(auton == -1)

{

auton = 2;

}

}

else if(controller->GetRawButton(5))

{

auton--;

Wait(.2);

if(auton == autonMax + 1)

{

auton = 0;

}

else if(auton == -1)

{

auton = 2;

}

}

auto str = std::to\_string(auton);

SmartDashboard::PutString("DB/String 5","Auton mode: " + str);

if(auton == 0)

{

SmartDashboard::PutString("DB/String 6","GOTTA GO FAST");

}

else if(auton == 1)

{

SmartDashboard::PutString("DB/String 6","OPEN THE BLAST DOORS");

}

else if(auton == 2)

{

SmartDashboard::PutString("DB/String 6","RIP SALLY");

}

else

{

SmartDashboard::PutString("DB/String 6","YOU MESSED UP");

}

}

void AutonomousInit()

{

driveTrainL1->SetControlMode(CANSpeedController::kFollower);

driveTrainL2->SetControlMode(CANSpeedController::kFollower);

driveTrainL3->SetControlMode(CANSpeedController::kFollower);

driveTrainL1->Set(0);

driveTrainL2->Set(0);

driveTrainL3->Set(0);

if(auton == 0)

{

driveTrainR1->Set(1);

driveTrainR2->Set(1);

driveTrainR3->Set(1);

Wait(5);

driveTrainR1->Set(0);

driveTrainR2->Set(0);

driveTrainR3->Set(0);

}

else if(auton == 1)

{

armExt1->Set(1);

Wait(1);

armExt1->Set(0);

RArm1->Set(1);

RArm2->Set(1);

Wait(1);

RArm1->Set(0);

RArm2->Set(0);

driveTrainR1->Set(1);

driveTrainR2->Set(1);

driveTrainR3->Set(1);

Wait(1.5);

driveTrainR1->Set(0);

driveTrainR2->Set(0);

driveTrainR3->Set(0);

}

else if(auton == 2)

{

RArm1->Set(1);

RArm2->Set(1);

Wait(1);

RArm1->Set(0);

RArm2->Set(0);

armExt1->Set(1);

Wait(1);

armExt1->Set(0);

driveTrainR1->Set(1);

driveTrainR2->Set(1);

driveTrainR3->Set(1);

RArm1->Set(-1);

RArm2->Set(-1);

Wait(1);

armExt1->Set(-1);

Wait(1);

armExt1->Set(0);

driveTrainR1->Set(1);

driveTrainR2->Set(1);

driveTrainR3->Set(1);

Wait(1);

driveTrainR1->Set(0);

driveTrainR2->Set(0);

driveTrainR3->Set(0);

}

}

void TeleopPeriodic()

{

TankDrive(leftJoy->GetRawAxis(1), rightJoy->GetRawAxis(1));

if(controller->GetRawButton(5))

{

SmartDashboard::PutString("DB/String 7","BEAM ME UP SCOTTY");

RArm1->Set(.5);

RArm2->Set(.5);

Wait(1);

RArm1->Set(0);

RArm2->Set(0);

armExt1->Set(1);

Wait(1);

armExt1->Set(-1);

Wait(1);

armExt1->Set(0);

}

}

void TankDrive(double left, double right)

{

driveTrainR1->Set(left);

driveTrainR2->Set(left);

driveTrainR3->Set(left);

driveTrainL1->Set(-right);

driveTrainL2->Set(-right);

driveTrainL3->Set(-right);

}

};

START\_ROBOT\_CLASS(Robot)