

```

// Harris Christiansen
// Created March 2014
// Team 3245
// HarrisChristiansen.com

package edu.frc.team3245;
import edu.wpi.first.wpilibj.*;
import edu.wpi.first.wpilibj.smartdashboard.SmartDashboard;

public class CompBot14 extends IterativeRobot {
    // Pilot Controls
    int leftStick=2, rightStick=4, fastBtn=8, slowBtn=7, lowGearBtn=5, highGearBtn=6;

    // Co Pilot Controls
    int togCompBtn=10, suckBtn=7, shootBtn=8, loadBtn=6, spitBtn=5, floorOutBtn=4,
    floorInBtn=2, spdUpBtn=3, spdDnBtn=1;

    // Motors
    private Talon leftMotor1, leftMotor2, leftMotor3, rightMotor1, rightMotor2, rightMotor3,
    shootMotor1, shootMotor2, floorMotor;

    // Current Motor Speeds
    private double leftSpeed, rightSpeed, shootSpeed1, shootSpeed2, floorSpeed;
    private double shooterSpeed=0.60;

    // Controllers
    Joystick pilotStick, coPilotStick;

    // Compressor
    Compressor mainComp;

    // Solenoids
    Solenoid driveHighSole, driveLowSole, floorOutSole, floorInSole;

    public void robotInit() {
        // Motors
        rightMotor1 = new Talon(4); // 4
        rightMotor2 = new Talon(5); // 5
        rightMotor3 = new Talon(6); //6
        leftMotor1 = new Talon(1); //1
        leftMotor2 = new Talon(2); //2
        leftMotor3 = new Talon(3); // 3
        shootMotor1 = new Talon(9); // 9
        shootMotor2 = new Talon(10); // 10
        floorMotor = new Talon(7); // 7

        // Joysticks
        pilotStick = new Joystick(1);
        coPilotStick = new Joystick(2);
    }
}

```

```

// Compressor
mainComp = new Compressor(14,1);

// Solenoids
driveLowSole = new Solenoid(1,1);
driveHighSole = new Solenoid(1,3);
floorOutSole = new Solenoid(1,5);
floorInSole = new Solenoid(1,7);

// Zero Motor Speeds
zeroMotorSpeeds();
}
private int autoConfig=0;
private boolean driveCopilot = false;
public void disabledPeriodic() {
    if(DriverStation.getInstance().getDigitalIn(1)) {
        autoConfig=1;
    }
    else if(DriverStation.getInstance().getDigitalIn(2)) {
        autoConfig=2;
    }
    else if(DriverStation.getInstance().getDigitalIn(3)) {
        autoConfig=3;
    }
    driveCopilot = DriverStation.getInstance().getDigitalIn(4);
}
private int autoCount=0;
public void autonomousInit() {
    zeroMotorSpeeds();
    shooterSpeed=0.645;
    autoCount=0;
}

public void autonomousPeriodic() {
    if(autoConfig==1) {
        zeroBallAuto();
    } else if(autoConfig==2) {
        oneBallAuto();
    } else if(autoConfig==3) {
        twoBallAuto();
    }
}

public void oneBallAuto() { // Drive and Shoot
    autoCount++;
    if(autoCount>=0&&autoCount<=75) {
        shootSpeed1=1.0;
        shootSpeed2=1.0;
    }
}

```

```

    leftSpeed=-0.35;
    rightSpeed=-0.375;
}
else if (autoCount>75&&autoCount<=270) {
    shootSpeed1=shooterSpeed;
    shootSpeed2=shooterSpeed;
    leftSpeed=0.0;
    rightSpeed=0.0;
}
else if (autoCount>270&&autoCount<=330) {
    shootSpeed1=shooterSpeed;
    shootSpeed2=shooterSpeed;
    floorSpeed=0.65;
    leftSpeed=0.0;
    rightSpeed=0.0;
} else {
    shootSpeed1=0.0;
    shootSpeed2=0.0;
    floorSpeed=0.0;
    leftSpeed=0.0;
    rightSpeed=0.0;
    if(autoCount==340) { mainComp.start(); }
}
updateMotors();
}

public void zeroBallAuto() { // Just Drive
    autoCount++;
    if(autoCount>=0&&autoCount<=100) {
        leftSpeed=-0.4;
        rightSpeed=-0.4;
    } else {
        leftSpeed=0.0;
        rightSpeed=0.0;
    }
    updateMotors();
}

public void twoBallAuto() { // 2 Ball - Shoot Drive Shoot
    autoCount++;
    if (autoCount>0&&autoCount<=50) {
        shootSpeed1=1.0;
        shootSpeed2=1.0;
        floorSpeed=0.0;
        leftSpeed=0.0;
        rightSpeed=0.0;
    } else if (autoCount>50&&autoCount<=100) {
        shootSpeed1=0.712;

```

```
    shootSpeed2=0.712;
    floorSpeed=0.0;
    leftSpeed=0.0;
    rightSpeed=0.0;
} else if (autoCount>100&&autoCount<=130) {
    shootSpeed1=0.712;
    shootSpeed2=0.712;
    floorSpeed=0.65;
    leftSpeed=0.0;
    rightSpeed=0.0;
} else if (autoCount>130&&autoCount<=200) { // Floor Out
    shootSpeed1=0.0;
    shootSpeed2=0.0;
    floorSpeed=0.65;
    leftSpeed=0.0;
    rightSpeed=0.0;
    floorOutSole.set(false);
    floorInSole.set(true);
} else if (autoCount>200&&autoCount<=210) { // Floor In
    shootSpeed1=0.0;
    shootSpeed2=0.0;
    floorSpeed=0.45;
    leftSpeed=0.0;
    rightSpeed=0.0;
    floorOutSole.set(true);
    floorInSole.set(false);
} else if (autoCount>210&&autoCount<=220) {
    shootSpeed1=1.0;
    shootSpeed2=1.0;
    leftSpeed=-0.35;
    rightSpeed=-0.375;
    floorSpeed=-0.45;
    floorOutSole.set(false);
    floorInSole.set(false);
} else if (autoCount>220&&autoCount<=230) {
    shootSpeed1=1.0;
    shootSpeed2=1.0;
    leftSpeed=-0.35;
    rightSpeed=-0.375;
    floorSpeed=-0.4;
} else if (autoCount>230&&autoCount<=285) {
    shootSpeed1=1.0;
    shootSpeed2=1.0;
    leftSpeed=-0.35;
    rightSpeed=-0.375;
    floorSpeed=0.0;
} else if (autoCount>285&&autoCount<=420) {
    shootSpeed1=shooterSpeed;
    shootSpeed2=shooterSpeed;
```

```

    leftSpeed=0.0;
    rightSpeed=0.0;
    floorSpeed=0.0;
} else if (autoCount>420&&autoCount<=480) {
    shootSpeed1=shooterSpeed;
    shootSpeed2=shooterSpeed;
    leftSpeed=0.0;
    rightSpeed=0.0;
    floorSpeed=0.65;
} else {
    shootSpeed1=0.0;
    shootSpeed2=0.0;
    floorSpeed=0.0;
    leftSpeed=0.0;
    rightSpeed=0.0;
    if(autoCount==490) { mainComp.start(); }
}
updateMotors();
}
public void teleopInit() {
    zeroMotorSpeeds();
    shooterSpeed=0.66;
    // Start Compressor
    mainComp.start();
}
public void teleopPeriodic() {
    updateTankDrive();
    updateDriveShifter();
    updateShooter();
    updateFloorLoader();
    updateToggleComp();
    updateMotors();
}
public void testInit() {

}
public void testPeriodic() {

}

// Custom Functions
public void zeroMotorSpeeds() {
    leftSpeed=0.0;
    rightSpeed=0.0;
    shootSpeed1=0.0;
    shootSpeed2=0.0;
    floorSpeed=0.0;
}

```

```

public void updateTankDrive() {
    double drivePercent = 0.55;
    if(pilotStick.getRawButton(fastBtn)) { drivePercent=1.00; }
    else if(pilotStick.getRawButton(slowBtn)) { drivePercent=0.30; }
    leftSpeed=pilotStick.getRawAxis(leftStick)*drivePercent;
    rightSpeed=pilotStick.getRawAxis(rightStick)*drivePercent;
    if(driveCopilot) {
        leftSpeed=coPilotStick.getRawAxis(leftStick)*drivePercent;
        rightSpeed=coPilotStick.getRawAxis(rightStick)*drivePercent;
    }
}

public void updateDriveShifter() {
    if(pilotStick.getRawButton(lowGearBtn)) {
        driveLowSole.set(true);
        driveHighSole.set(false);
    } else if(pilotStick.getRawButton(highGearBtn)) {
        driveLowSole.set(false);
        driveHighSole.set(true);
    } else {
        driveLowSole.set(false);
        driveHighSole.set(false);
    }
}

public boolean spdCngd=false;
public int shootCount=0;
public void updateShooter() {
    if(coPilotStick.getRawButton(shootBtn)) {
        shootCount++;
        if(shootCount<60) {
            shootSpeed1=1.0;
            shootSpeed2=1.0;
        } else {
            shootSpeed1=shooterSpeed+0.1;
            shootSpeed2=shooterSpeed-0.1;
        }
    }
    if(mainComp.enabled()) { // FOR TOMORROW - Test If Working
        mainComp.stop();
    }
}
else if(coPilotStick.getRawButton(suckBtn)) { shootSpeed1=-0.41; shootSpeed2=-0.41; }
else {
    shootCount=0;
    if(!mainComp.enabled())&&shootSpeed1!=0.0) { // FOR TOMORROW - Test If Working
        mainComp.start();
    }
    shootSpeed1=0.0;
    shootSpeed2=0.0;
}

```

```

    }
    if(coPilotStick.getRawButton(spUpBtn)&&!spdCngd) { spdCngd=true;
shooterSpeed=shooterSpeed+0.005; System.out.println(shooterSpeed); }
    else if(coPilotStick.getRawButton(spDnBtn)&&!spdCngd) { spdCngd=true;
shooterSpeed=shooterSpeed-0.005; System.out.println(shooterSpeed); }
    if(!coPilotStick.getRawButton(spUpBtn)&&!coPilotStick.getRawButton(spDnBtn))
{ spdCngd=false; }
    SmartDashboard.putNumber("Shooter Speed", shooterSpeed);
}

```

```

public void updateFloorLoader() {
    if(coPilotStick.getRawButton(loadBtn)) { floorSpeed=0.8; } // 65
    else if(coPilotStick.getRawButton(spitBtn)) { floorSpeed=-0.8; }
    else { floorSpeed=0.0; }
    if(coPilotStick.getRawButton(floorOutBtn)) {
        floorOutSole.set(true);
        floorInSole.set(false);
    } else if(coPilotStick.getRawButton(floorInBtn)) {
        floorOutSole.set(false);
        floorInSole.set(true);
    } else {
        floorOutSole.set(false);
        floorInSole.set(false);
    }
}
}

```

```

boolean togCompSwitched=false;
public void updateToggleComp() { // FOR TOMORROW - test if working
    if(coPilotStick.getRawButton(togCompBtn)&&!togCompSwitched) {
        togCompSwitched=true;
        if(mainComp.enabled()) {
            mainComp.stop();
        } else {
            mainComp.start();
        }
    } else {
        togCompSwitched=false;
    }
}
}

```

```

public void updateMotors() {
    double correction = 11.2/DriverStation.getInstance().getBatteryVoltage();
    if(DriverStation.getInstance().getBatteryVoltage()<8.00) {
        correction = 1.00;
    }

    leftMotor1.set(leftSpeed*correction);
    leftMotor2.set(leftSpeed*correction);
    leftMotor3.set(leftSpeed*correction);
}

```

```
rightMotor1.set(-rightSpeed*correction);
rightMotor2.set(-rightSpeed*correction);
rightMotor3.set(-rightSpeed*correction);

shootMotor1.set((-shootSpeed1*correction));
shootMotor2.set((-shootSpeed2*correction));
SmartDashboard.putNumber("Shooter Voltage",
shooterSpeed*correction*DriverStation.getInstance().getBatteryVoltage());
floorMotor.set(floorSpeed*correction);
}
}
```