TEAM 6346

CYBEARS

FRC 2024

Robot Code Documentation:

The purpose of this document is to easily modify the robot code at competition without breaking the code. Future revisions will include complete explanations of the program's structure and functions.

Table of Contents

Teleop Drive Speed

Driving.java class >> ManualDrive function >> Line 36 >> double speedPercentage = 0.7; //range 0.0 to 1.0

- Default = 0.7
- Full Speed = 1.0
- Tight corridor = 0.55

Teleop Arm Speed (*Recommend DO NOT change)

Arm.java class >> SetSpeed function >> Line 47 >> double speedPercentage = 0.3; //range 0.0 to 1.0

- Default = 0.3
- Max Safe Speed = 0.5

NOTE: Changes here will affect AUTO Arm Speed

Autonomous Shooting Arm Position

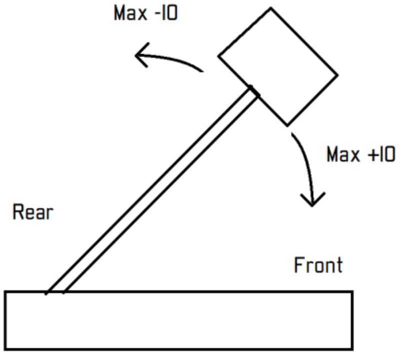
This is the arm position that the arm will attempt to shoot at during autonomous. Adjust this value until consistently shooting into the Center speaker position.

Test this using the Stationary Score autonomous selection

Arm.java class >> zeroArm function >> line 71 >> double ShootPosition_Offset = 0; //Positive = Lower Arm, Negative = Raise Arm (+/- 10)

- Default = 0
- Lowest Arm = +10
- Highest Arm = -10

Note: These limits can be exceeded but it is unlikely you will need to adjust higher or lower.



Autonomous Movement Arm Position

This is the arm position that the arm will move to after shooting and potentially picking up a ring in autonomous. This position is off the ground but still keeping the weight to the front.

Test this using the Stationary Score autonomous selection.

Autonomous.java class >> line 11 >> double MovementPosition_Offset = 34; //Positive = Lower Arm, Negative = Raise Arm. Range (-4 to +16)

- Default = 0
- Lowest Arm = 16
  *Don't know if this will hit the front Limit Switch
- Highest Arm = -4
  *No real benefit being higher since that is practically shooting position

Note: These limits can be exceeded but it is unlikely you will need to adjust higher or lower.

Amp Arm Position

This is the arm position that will display a green light on the shuffleboard when the arm is in the correct position to score at the Amp. This can be adjusted on the fly in Teleop by pressing the Back button on the Arm Controller when in the correct position.

Test this using the Stationary Score autonomous selection then checking the Shuffleboard Arm Position box in Teleop.

Arm.java class >> zeroArm function >> line 72 >> double AmpPosition_Offset = 0; //Positive = Lower Arm, Negative = Raise Arm (+/- 10)

- Default = 0
- Lowest Arm = +10
- Highest Arm = -10

Note: These limits can be exceeded but it is unlikely you will need to adjust higher or lower.

Debug Testing Individual Motor

This is used to quickly test if a motor or Spark Max is faulty. The code is prewritten but commented out. To remove the comments, delete the / and */ at lines 108 and 113.*

NOTE: This code MUST be commented out for Teleop to work correctly. So replace the /* and */ after testing

Robot.java class >> teleopPeriodic function >> line 108 – 113 >>

```
//Test Individual Motor Code: Controlled by Y button on the Driver Controller
If (DriverController.getYButton() == true){
        TestMotor(0); //Change TestMotor(CAN_ID) to desired motor. Range 1 to 13
}
```

- Drive Motors = 1-4
- Arm Motors = 5-6
- Winch Motors = 7-8
- Pickup Motor = 9
- Shooter Motors = 10-13

Autonomous Preset_Move

This is how to add a Preset_Move command to an autonomous mode.

```
//Preset_Move
DistanceFT = 0; //Positive = Forward, Negative = Reverse. Actually pretty close to number of feet moved
wheelSpeed = 0.25; //Do NOT change
SmartDashboard.putString("Auto Command", "Preset Move Forward/Reverse");
Preset_Move(DriveSub, DistanceFT, wheelSpeed, 0);
```

Autonomous Preset_Pickup

This is how to add a Preset_Pickup command to an autonomous mode.

```
//Preset_Pickup
DistanceFT = 0; //Always Positive. Actually pretty close to number of feet moved
wheelSpeed = 0.25; //Do NOT change
SmartDashboard.putString(“Auto Command”, “Preset_Pickup”);
Preset_Pickup(PickupSub, DriveSub, DistanceFT, wheelSpeed);
```

Note: If you want to set arm to Front Limit Switch, save time by lowing it while moving.
Before the //Preset_Pickup, add **ArmSub.SetSpeed(-0.6)**

Autonomous Preset_Turn

This is how to add a Preset_Turn command to an autonomous mode.

```
//Preset_Turn
TurnTune = 0;
SmartDashboard.putString(“Auto Command”, “Preset_Turn Direction & Angle”);
Preset_Turn(DriveSub, direction, angle, TurnTune);
//If Overturning, set TurnTune between -1 and -20
//If Underturning, set TurnTune between 1 and 20
```

- **direction** = “LEFT” or “RIGHT” and must be ALL CAPS and in quotes “ ”
- **angle** = 90 or 60

Autonomous Preset_Aim

This is how to add a Preset_Aim command to an autonomous mode.

```
//Preset_Aim
SmartDashboard.putString(“Auto Command”, “Preset_Aim to X Position”);
Preset_Aim(ArmSub, Position, true);
```

- **Position** options
  - ArmSub.getZero()+MovementPosition
    - Moves arm to the Movement Position
  - ArmSub.getShootPosition()
    - Moves arm to the Shooting Position
  - ArmSub.getAmpPosition()
    - Moves arm to the Amp Position
  - ArmSub.getZero()
    - Moves arm to Rear Limit Switch (straight up)

Note: If you want to move the arm to the Front Limit Switch see note in Preset_Pickup

Auto_StationaryScore (*Recommend DO NOT change)

Autonomous.java class >> Auto_StationaryScore *function >> Line 27*
This autonomous will be run first for every autonomous mode.

- Move arm to Shooting Position
    - Line 30
- Fire the Shooter
    - Line 33
- Move arm to Movement Position
    - Line 36

Auto_Source

Autonomous.java class >> Auto_Source function >> Line 40
This autonomous has two variants for Red and Blue alliance.

- Run Auto_StationaryScore
    - You can adjust **armTune** to adjust the shooting angle for just this autonomous mode
        - Line 45
- Move forward 15.6 ft
    - Line 52
- Turn 60 degrees:
    - Red = RIGHT
        - Line 55 to 62
    - Blue = LEFT
        - Line 63 to 70
- Lower arm to Front Limit Switch
    - Line 73
- Move forward 17.8 ft and run Pickup
    - Line 79

Note: I have included commented code that should only be enabled after the ring is successfully picked up

- Run Pickup and Lift Arm to Movement Position
    - Lines 85 to 89
- Move Reverse -15ft
    - Lines 92 to 96

Auto_Center

Autonomous.java class >> Auto_Center function >> Line 102
This autonomous has two variants for Red and Blue alliance but is currently the same for both.

- Run Auto_StationaryScore
    - You can adjust **armTune** to adjust the shooting angle for just this autonomous mode
        - Line 107
- Lower arm to Front Limit Switch
    - Line 111
- Move forward 7 ft and run Pickup
    - Line 117
- Run Pickup and Lift Arm to Movement Position
    - Lines 120 to 123
    - This can be safely ignored or commented out if no teammate wants to leave a ring for us to pickup

Note: I have NOT added any code after a successful pickup. Here are my recommendations

- Preset_Aim to ShootPosition + SomeNumber and attempt a shot from your current position
- Preset_Move Reverse (–DistanceFT) to the speaker >> Run Auto_StationaryScore

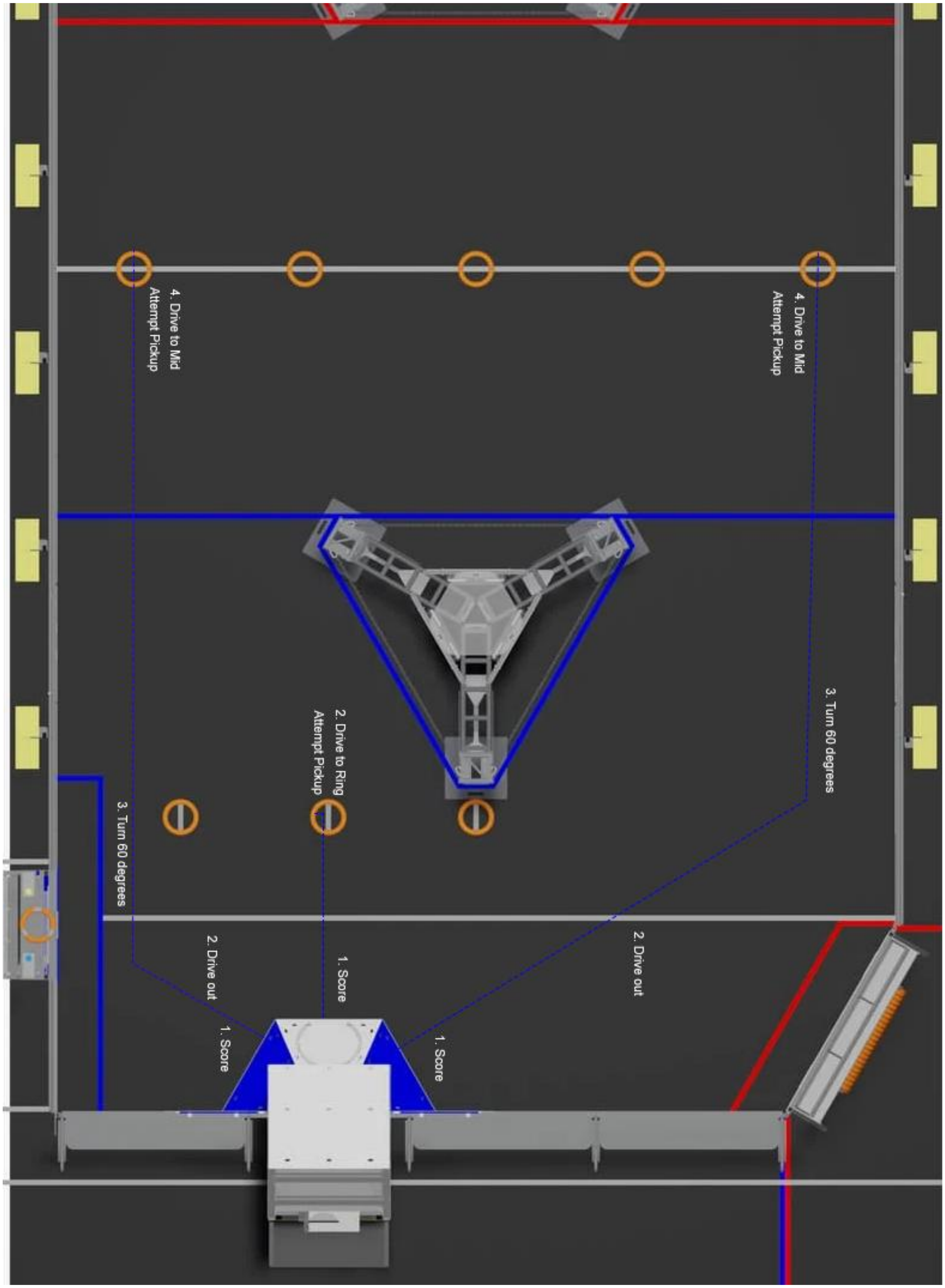
Auto_Amp

Autonomous.java class >> Auto_Amp function >> Line 133
This autonomous has two variants for Red and Blue alliance.

- Run Auto_StationaryScore
    - You can adjust **armTune** to adjust the shooting angle for just this autonomous mode
        - Line 136
- Move forward 4.6 ft
    - Line 145
- Turn 60 degrees:
    - Red = LEFT
        - Line 148 to 155
    - Blue = RIGHT
        - Line 156 to 165
- Lower arm to Front Limit Switch
    - Line 166
- Move forward 23.6 ft and run Pickup
    - Line 172

Note: I have included commented code that should only be enabled after the ring is successfully picked up

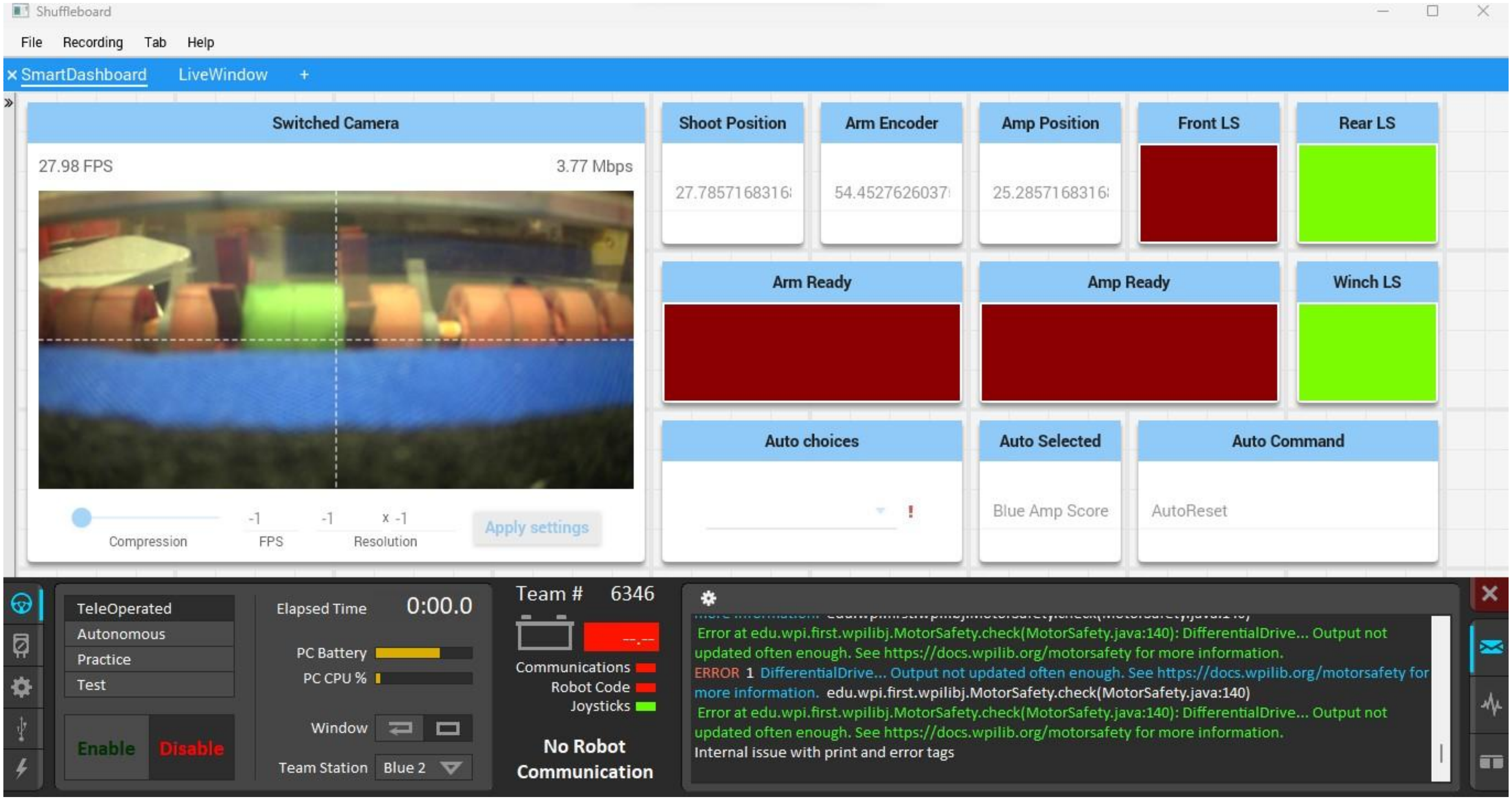
- Run Pickup and Lift Arm to Movement Position
    - Lines 178 to 182
- Move Reverse -15ft
    - Lines 185 to 189



4. Drive to Mid
Attempt Pickup
4. Drive to Mid
Attempt Pickup
3. Turn 60 degrees
2. Drive to Ring
Attempt Pickup
3. Turn 60 degrees
2. Drive out
1. Score
2. Drive out
1. Score
1. Score

IMPORTANT NOTES:

- Adjust the winches to be just above, but not on, the Winch Limit Switch when setting up the robot on the field
    - Putting the winches all the way down could break the Limit Switch
    - Putting the winches all the way up will take up more time to lower and zero the encoder
- If you ever need to override the winch encoder limit for extending:
    - Hold A button on the Driver Controller while using the Triggers to adjust the Winch
- After running Autonomous, you must enable and disable Teleop before you can run Autonomous again
    - This will be confirmed in the Auto Command box on the shuffleboard reading "AutoReset"
    - Alternatively, you can redeploy the code
- Shuffleboard can sometimes fail to display the Auto Choices dropdown menu
    - Make sure to connect to the robot and open shuffleboard before powering down to setup on the field.
- If the Winch or Arm jams during the zeroing at the beginning of Autonomous
    - Press the Auto E-Stop
    - Normally need to redeploy code but you cannot do this when connected to the Field Management System.



Switched Camera: 27.98 FPS, 3.77 Mbps; -1 FPS; -1 x -1 Resolution

Shoot Position: 27.785716831б

Arm Encoder: 54.452762603

Amp Position: 25.285716831б

Front LS: 🟥

Rear LS: 🟩

Arm Ready: 🟥

Amp Ready: 🟥

Winch LS: 🟩

Auto choices: !

Auto Selected: Blue Amp Score

Auto Command: AutoReset

Elapsed Time 0:00.0

PC Battery

PC CPU %

Window

Team Station Blue 2

Team # 6346

Communications 🟥

Robot Code 🟥

Joysticks 🟩

No Robot Communication

```
 Error at edu.wpi.first.wpilibj.MotorSafety.check(MotorSafety.java:140): DifferentialDrive... Output not
updated often enough. See https://docs.wpilib.org/motorsafety for more information.
ERROR  1  DifferentialDrive... Output not updated often enough. See https://docs.wpilib.org/motorsafety for
more information.  edu.wpi.first.wpilibj.MotorSafety.check(MotorSafety.java:140)
 Error at edu.wpi.first.wpilibj.MotorSafety.check(MotorSafety.java:140): DifferentialDrive... Output not
updated often enough. See https://docs.wpilib.org/motorsafety for more information.
Internal issue with print and error tags
```