

# Pinewood Derby Manual

---

## (AWANA Grand Prix)

Copyright © 2022 Randix LLC. All rights reserved.

## Introduction

This app will set up the data and operate the running of a Pinewood Derby / Grand Prix electronic timer for a race. The software consists of the iOS App, available on the Apple AppStore, and a Python based timer interface. The Python software can be found in the application directory.

The complete system consists of:

- at least one iPhone or iPad device (multiple devices may be used by observers),
- (currently) a **microwizard.com** Fast Track timer device with power supply,
- a computer which can run Python (Windows, Linux, Mac, **Raspberry Pi**, etc.),
- an RS232 serial to USB cable,
- and a switch cable (RJ22 connectors) to the start switch.

### iOS Pinewood-Derby App

The iOS **Pinewood-Derby** App, available in the Apple AppStore, has two modes, a administrator mode and an observer mode.

The observer mode allows:

- Viewing of the racers.
- Viewing of the heats.
- Viewing of the times.
- Viewing of the rankings.
- Viewing of the results.
- Connecting to the computer (e.g., Raspberry Pi) timer server.
- Getting the race configuration from the Raspberry Pi timer server.

The administrator mode allows full control of the race data, including

- Everything that the observer mode can do, plus:
- Adding, editing, and deleting groups and racers.

- Managing the heats of the race, including entering special heats.
- Editing the resulting times, to allow ignoring false times.
- Sending the configuration to the computer (e.g., Raspberry Pi) timer server.
- Start and resume Racing.
- Start and resume Simulation (for testing).

### Computer (e.g., RaspberryPi) Timer Connection

This will assume in this section that the computer used is a Raspberry Pi, RPi. Pretty much any computer may be used to attach to the timer and send the data to the iPhones and/or iPads. This document will address setting up on the RPi, which extends to most Linux computers, an Apple Mac computer (MacBook, etc.), and a Microsoft Windows computer.

The RPi and the iPhones and iPads must be connected to the local WiFi in order to communicate.

## Technical Detail

The timer computer is started and connected to the local WiFi. This will be discussed below in Installation. the file PIN.txt is edited to create a PIN for the iOS application. This ensures that only the race administrator can have access to the administration mode. Then the programs (applications) ***PDServer.py*** and ***timer.py*** are started on the timer computer.

The timer computer is connected physically via serial cable to the timer and makes the timing data available over WiFi, the iPads and iPhones communicate with the timer computer to fetch data, as well as the administrator informs the timer computer of which cars are racing in a given heat.

## Installation

Hardware set-up:

- Assemble the race track with the timer.
- Connect the start switch cable to the track start switch.
- Connect the serial cable from the track to the RS232 to USB converter cable and plug that into the timer computer.
- Start the timer computer
  - Assure that the timer computer is connected to the local WiFi. The timer computer makes no use of any available internet connection.

- Open two windows on the timer computer, using the Terminal program.
- Install the **PDTimer.tar** Python programs. See detail below in *Appendix: Timer Server Software Installation*.
- Edit the file *PIN.txt* and enter a four digit PIN to be used by the administrator during the race, select a secret pin, known only to the administrator. E.g.: `echo 4567 > PIN.txt`
- On a Raspberry Pi, other Linux computer or a Mac computer, start the software:
 

```
./t.sh
```
- On a Windows timer computer, assuming Python3 has been installed, see below in *Appendix: Timer Server Software Installation*.
  - In one window start the **PDServer.py** with the command `Python3 PDserver.py`
  - In the other window start the **timer.py** with the command `Python3 timer.py`
- Start **Pinewood-Derby** on the iPhone or iPad
  - Tap on Settings in the upper right corner.
  - Enter the PIN that was set in the file *PIN.txt* on the RPi.
  - You may now set the Title and the Event.
  - Ascertain that the number of tracks set is correct.
  - Swipe down on the Settings page from the top, or tap the *Dismiss* button.
- Select the Racers tab and enter all racers, car numbers, car names, and ages.
  - Swipe left on a racer entry to delete or edit the entry.
  - Tap on the Add button to enter a new racer.
  - The Groups list may be edited with the Groups button.
- When the racers are all correctly entered, tap on the Settings again.

Ascertain that the app is connected to the RPi server, the Connected flag must be a green checkmark. If it is a red X, see the fine print, and solve the connection problem.

This is the connection to the RPi timer, and must work correctly to get the times from the timer.

To Start the race, tap on the Racing: Start button. This will take you to the Heats tab, where the individual heats may be started.

## Running the Race Heats

To start a heat, tap on the heat. This will pop-up a view of which cars should be placed in which tracks. When the cars are ready, tap on Start.

The track should be setup to place the cars on the tracks. When ready, release the cars, this will start the timer. When the cars go through the timer, the timer will record the times of each car, as well as the finish order.

**Pinewood-Derby** will read the times from the timer, and display them in the Heats view, as well as marking the heat gray, indicating that it has been run.

## Pinewood-Derby Views

There are five different views in the application found by the buttons on the bottom as well as a Settings view available by the button in the upper right corner.

Generally all columns will be sorted by that column by tapping on the column header button.

### Settings

This view allows changing the access string for the Timer. This is normally

```
https://raspberrypi.local:8484/
```

This can be changed to whatever computer you may be using, if not using a Raspberry Pi as described below in the Appendix.

The application will only work if it is connected to the Timer computer and the Connected green checkmark is displayed.

The administrator of the race should enter the PIN for additional settings and race control. Be very careful with the administrator role. There should probably be only one race administrator. The data is not kept

### Administrator View

For the case that you are displaying the race on a large screen, you may change the Title and Event.

Select the number of physical tracks being used. DO NOT CHANGE this during a race, all heat times may be lost.

You can start the race with Race: **Start**.

The **Resume** button is for the case that the application encounters a software error and crashes, it can be restarted and resumed without losing the previously accumulated heat runs.

For demonstrating how the application operates without being connected to a Timer computer, you can use the Simulation Testing: **Start** and **Resume**.

## **Racers**

This displays the cars and racers.

Racers may be added with the Add button in the upper left (for administrators). Already entered racers can be edited or deleted by swiping left on an entry and selecting the red Trash can or the turquoise Edit buttons.

## **Heats**

This displays the list of heats (these are calculated at the start of the race). The heat generation rule is: each car should run once in each track.

Tap on a heat to begin the process of starting that heat. The entry will display the winning position and the times for each racer, and the entry will be gray when the heat has run.

Special Heats may be created using the **Special** button in the upper left corner.

## **Times**

This displays each car's times and positions, as well as a sum of the position placings and the average times.

Times may be ignored by swiping left on an entry and selecting an invalid time to ignore.

## **Rankings**

This displays the rankings. This can be very useful by sorting on any column to see the desired ordering.

## **Results**

This displays the final results. This is a redundant display of the Rankings. This shows the final winners and slowest cars.

## **Appendix: Timer Computer and Cable Installation**

The Timer must be connected to the track start switch.

The Timer must be connected to the Timer Computer via a serial cable, and a serial to USB converter.

## Appendix: Timer Server Software Installation

The Timer computer software is distributed automatically with the **Pinewood-Derby** app. It can be found after installation and first start of the app using the **Files** app and navigating to:

### On My iPad or On My iPhone → Pinewood-Derby

There is a file **PDServer.tar** to found there.

The files may also be found at:

[GitHub - randix/Pinewood-Derby: Time and Score a Pinewood Derby race.](#)

This file will be transfered to the Timer computer (preferably a RaspberryPi). There are many ways to do this. This will describe the process for Raspberry Pi, or a Mac computer. This can also be done on a MS Windows computer, but the details here will be somewhat sketchy.

#### 1. Copy PDServer.tar to the Timer Computer

In general, you will find an app on the Apple AppStore that can transfer a file from the iPhone/iPad to the Timer computer. One (of many such) apps, if the Timer computer is a Raspberry PI or a Mac computer, is

**FTPManager - FTP, SFTP client** ( <https://apps.apple.com/us/app/ftpmanager-ftp-sftp-client/id525959186> ).

#### 2. Install PDServer.tar on the Timer Computer

Log in to the Timer computer. Issue the following commands:

```
mkdir PDServer
mv PDServer.tar PDServer
cd PDServer
tar xf PDServer
```

#### 3. Create the PIN.txt File and Run the PDServer on the Timer Computer

In the terminal window from above:

```
./timer.py
```

Open a second terminal window and issue these commands, selecting a desired PIN in the place of "1234":

```
cd PDServer  
echo 1234 > PIN.txt  
./PDServer.py
```

---

**Contact:**

Randix LLC

[rand@randix.net](mailto:rand@randix.net)