Pinewood Derby Tools

Timer Software

This page provides information to aid in the setup/configuration of the timer sketch for the Pinewood Derby Timer (PDT) including interfacing the timer to race management software running on a PC. Refer to the Construction page for information on constructing the timer's hardware (circuit, bridge, etc.).

The following subjects are covered:

Arduino Environment

The Arduino environment consists an integrated development environment (IDE) that allows programs to be edited, compiled and uploaded to the microcontroller (board). Refer to the Arduino web page which corresponds to your system type and follow the steps to download, install and test the IDE:

- Windows: https://www.arduino.cc/en/Guide/Windows
- Mac OS: https://www.arduino.cc/en/Guide/MacOSX
- Linux: https://www.arduino.cc/en/Guide/Linux

STOP Do not continue until you have successfully uploaded the blink example to your board as described in the above instructions. If you could not get the blink example to work you will not be successful with the PDT sketch.

Adafruit Display Libraries

If your PDT build will include the optional Adafruit 7 segment displays you will need to install the libraries needed to drive the displays. Information about installing libraries can be found on the following web page: https://www.arduino.cc/en/Guide/Libraries

Referring to the first section, "How to Install a Library using the Library Manager", search for and install the following two libraries:

- Adafruit GFX Library
- Adafruit LED Backpack Library

PDT Sketch

If you have not done so already download the PDT sketch (CHANGELOG). Create a folder (it can be anywhere on your computer) named **timer** and place the PDT sketch, named **timer.ino,** into the folder.

There are a couple of quirks in the Arduino environment:
1) the folder name MUST match the name of the PDT sketch. You can change the name but they must match. So, if you name the sketch **fred.ino** then the name of the folder must be **fred.**
2) Keeping multiple copies of the PDT sketch (backups for example) in the folder will cause the compile to fail. There should only be one sketch file in the folder, move any backups to another directory.

BACK TO TOP

Timer Configuration

In the Arduino software open the PDT sketch timer.ino that you downloaded above. This section will describe the available configuration options for the timer.

The top of the PDT sketch contains the most common options that can be configured:

```
/*-----------------------------------------*
 - TIMER CONFIGURATION -
 *-----------------------------------------*/
#define NUM_LANES      1           // number of lanes
#define GATE_RESET     0           // Enable closing start gate to reset timer

//#define LED_DISPLAY    1           // Enable lane place/time displays
//#define DUAL_DISP      1           // dual displays per lane (4 lanes max)
//#define DUAL_MODE      1           // dual display mode
//#define LARGE_DISP     1           // utilize large Adafruit displays (see website)

#define SHOW_PLACE     1           // Show place mode
#define PLACE_DELAY    3           // Delay (secs) when displaying place/time
#define MIN_BRIGHT     0           // minimum display brightness (0-15)
#define MAX_BRIGHT     15          // maximum display brightness (0-15)
/*-----------------------------------------*
 - END -
 *-----------------------------------------*/
```

Changing Number of Lanes

The default version of the PDT is configured for a one lane track. If you have a multiple lane track you will need to update the PDT sketch. The NUM_LANES variable should be changed from the default value of 1 to match the number of lanes on your track:

```
#define NUM_LANES      1          // number of lanes
```

Optional Adafruit Displays

The next five lines are used to configure the optional Adafruit displays. The default setup of the PDT is configured to not have any displays – the results are only sent to an attached PC. If you choose to utilize the optional lane place/time displays, use the following parameters to enable and configure them:

Enable Displays: The Adafruit libraries must be installed as described in the Arduino environment section above. To enable the displays the following line needs to be uncommented (remove the two slashes at the beginning of the line) so that it changes from this:

```
//#define LED_DISPLAY    1           // Enable lane place/time displays
```

to this:

```
#define LED_DISPLAY    1           // Enable lane place/time displays
```

Display Behavior: By default, when utilizing the display functionality, as cars finish the race their finishing place is displayed. When all cars have finished the displays will then alternate between the finish order and elapsed time. To change the time delay between the two, change the following parameter to desired number of seconds:

```
#define PLACE_DELAY    3           // Delay (secs) when displaying time/place
```

If it is desired to only show elapsed time – with no finish order – the following line should be changed from the default,

```
#define SHOW_PLACE     1           // Show place mode
```

to this:

```
#define SHOW_PLACE     0           // Show place mode
```

Display Brightness: The following two parameters limit the range of the dimmer knob which controls the brightness of the lane displays. Limiting the maximum brightness can control (usually to minimize) the total power draw of the lane displays.

```
#define MIN_BRIGHT     0           // minimum display brightness (0-15)
#define MAX_BRIGHT     15          // maximum display brightness (0-15)
```

Advanced Display Configurations: The following two parameters enable two advanced display configurations. They are enabled by uncommenting the lines as described above for the LED_DISPLAY parameter.

```
//#define DUAL_DISP      1           // dual displays (front/back) per lane (4 lanes max)
//#define DUAL_MODE      1           // dual display mode
//#define LARGE_DISP     1           // utilize large Adafruit displays (see website)
```

The first, DUAL_DISP, is a feature to drive two displays per lane – one on the front and one on the rear of the finish bridge. This feature is only available for configurations with 4 or less lanes. The paired front/back displays will show identical information. Refer to the Construction page for details on addressing the displays for this feature.

The second, DUAL_MODE, is used in conjunction with the DUAL_DISP option described above. When enabled (by removing the 2 leading slashes), the secondary display must be a Adafruit 8x8 matrix display and it will show the place order of the lane while the primary display shows the time. Like the normal dual display option, this feature is only available for configurations with 4 or less lanes. The 8x8 matrix displays are addressed the same as the normal dual displays, refer to the Construction page for details. **NOTE!** For this mode to operate correctly, the SHOW_PLACE option (described previously) must be set to 0. The final configuration of these three parameters for this feature would be:

```
#define SHOW_PLACE     0          // Show place mode
#define DUAL_DISP      1          // dual displays (front/back) per lane (4 lanes max)
#define DUAL_MODE      1          // dual display mode
```

The third, LARGE_DISP, is a feature to drive the larger (1.2") Adafruit displays. Because of the configuration of the displays – specifically the decimal placement – the displays should be mounted upside down. In addition to enabling the feature in the PDT the Adafruit library must be manually modified – refer to the instructions on Construction page for details.

Gate Reset: The following parameter (when set to 1) enables the gate reset functionality, the default is disabled (set to 0). When enabled the timer – when in the FINISHED state – can be reset by closing the start gate. Caution should be taken if manually recording results – resetting the timer clears the displays.

```
#define GATE_RESET     0          // Enable closing start gate to reset timer
```

Status Light (LED) Troubleshooting: During testing or operation of the timer you might notice that the green and blue status colors seem to be swapped This can happen because the supplier of the LED keeps switching the source part and the pin orders on the components vary.

This can be fixed by changing the 3 lines in the "pin assignments" section of the sketch from,

```
byte STATUS_LED_R = 9;            // status LED (red)
byte STATUS_LED_B = 10;           // status LED (blue)
byte STATUS_LED_G = 11;           // status LED (green)
```

to the following (the only change is swapping the '10' and '11'),

```
byte STATUS_LED_R = 9;            // status LED (red)
byte STATUS_LED_B = 11;           // status LED (blue)
byte STATUS_LED_G = 10;           // status LED (green)
```

⚠ Don't forget to save your changes!

BACK TO TOP

Uploading Timer Sketch

Once the timer configuration is completed upload the PDT sketch to the board by pressing the "Upload" button in the Arduino software. This is the same process used when you tested your environment by uploading the blink example earlier.

BACK TO TOP

Hardware Test Mode

When utilizing the LED displays on the PDT a hardware test mode is available to help test and verify the PDT construction and operation.

Enter Test Mode: While pressing (and holding) the PDT reset switch plug in the PDT (or hit the Arduino reset button). If the status LED is lit it will go dark. When the status LED lights release the PDT reset switch. The 3 stages of the test mode are described below. To move to the next stages, press the PDT reset switch until the displays blank and then release the PDT switch. The status LED will stay lit.

Lane Detector Test Stage: Each lane's display will indicate whether the timer is reading the lane as dark (== d) as if a car is crossing the finish line or as lit (== L) as if no car is present.

Start Gate Test Stage: The first lane's display will indicate whether the timer is reading the start gate as closed (S=CL) as if the cars are staged or as open (S=OP) as if the cars have been released.

Display Brightness Stage: Each lane's display will display the currently set brightness level.

BACK TO TOP

Timer Operation

Operation of the PDT is simple, the current status of the timer is indicated by the color of the status LED.

READY (Blue LED): The timer is ready for a race to start. If displays are used, they will each show dashes '----' indicating the PDT is ready.

Hitting the reset button, the PDT (with the start gate closed) puts the timer into the 'READY' state.

Starting the race by opening the start gate (either via software or manually) causes the timer to leave the 'READY' state and enter the 'RACING' state.

RACING (Green LED): The race has started and the timer is watching each lane for cars to finish. If displays are used, they will each go blank indicating the race is in progress.

If displays are used, as cars finish the displays will show either their current place (1st, 2nd, etc.) or their elapsed time (in seconds) depending on how the timer was configured.

Once all cars have finished the timer will leave the 'RACING' state and enter the 'FINISHED' state.

Hitting the reset button while in the 'RACING' state will cause the current race to end and will put the timer in the 'FINISHED' state.

FINISHED (Red LED): The race is over. If displays are used, they are either alternating between place and time or simply showing the time (based on the timer configuration).

Hitting the reset button (with the start gate closed) will cause the timer to leave the 'FINISHED' state and enter the 'READY' state. If the start gate is open when the reset button is hit the timer will remain in the 'FINISHED' state. If the Gate Reset option is enabled the timer will be reset automatically when the start gate is closed.

⚠ If the blue and green status colors seem to be swapped see the "Status Light (LED) Troubleshooting" information in the earlier Timer Configuration section.

BACK TO TOP

Interface to Race Management Software

PDT is supported by the following race management applications:

- GrandPrix Race Manager
- Pinewood Master
- DerbyNet

GrandPrix Race Manager

The Grand Prix Race Manager (GPRM) software is available on Windows or Mac.

To configure GPRM for use with the PDT within GPRM select the "Settings" menu and then select "Hardware Settings..."...". On the resulting "Hardware Settings" window on the "Timer tab", select "Other Serial Timer" in the "Timing System" group then hit the "Custom Settings" button. On the resulting "Custom Timer Settings" window enter the following configuration and then hit the "Save" button.

```
COM Port Settings
  Baud:                    9600
  Data Bits:               8
  Parity:                  None
  Stop Bits:               1

Start Gate Status
  Check Command:           G
  Open Response:           O
  Timer Start Message:     B

Timer Reset
  Reset Command:           R
  Ready Response:          K
  Response Delay:          0.25 secs

Lane Masking
  Mask Command:            M
  Remove all Masks:        U

Other Software Commands
  Retrieve Data:           LEAVE THIS FIELD BLANK!
  Force Data Send:         F
  Trigger Solenoid:        S

Miscellaneous
  Read Delay:              0.25 secs
  Precision:               0.0001 secs
  Max Time:                9.9999 secs
  Lane Labels:             Numbers
```

Now select the "Settings" menu and then select "Software Settings...". On the resulting "Software Settings" window on the "Track" tab, set the number of lanes in the "Lanes" group to match the number of lanes on your track and then hit the "Save" button.

Refer to the GPRM help for general usage of the application.

Pinewood Master

The Pinewood Master software is available on Mac. The Pinewood Master website has detailed instructions on how to configure the software to work with the PDT.

DerbyNet

The DerbyNet software is available on Windows, Mac or Linux. The DerbyNet website has instructions on how to configure the software to work with the PDT.

BACK TO TOP

PD Test/Tune/Track (PDT3) Software

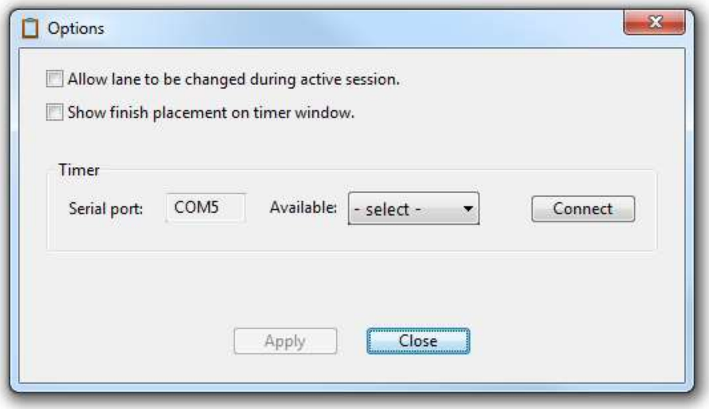
The PDT3 utility is a free software program that fully interfaces with the PDT and allows you to record, analyze and archive car run times during the testing/tuning process of building a Pinewood Derby car.

> ⚠ If you experience issues with the PDT3 utility running on Windows 10 systems you will need to install the latest Windows updates. USB communication issues have been fixed in these updates. (October, 2016)

Configuration

After installation a shortcut for PDT3 will be placed on your desktop. If needed the PDT3 utility can be uninstalled via the Control Panel → Programs and Features window.

The first step to perform after installation is to setup the connection between the PDT and the PDT3 utility. Selecting "Options..." from the "Edit" menu will display the following dialog:



Options

□ Allow lane to be changed during active session.

□ Show finish placement on timer window.

Timer

Serial port: COM5 Available: - select -

> ⚠ You might experience a long delay when this window is opened or when you attempt to select a COM port. This is caused by a bug in the programming language that was used to create this utility. The window will eventually respond and return to normal.

In the "Timer" group select the COM port for the PDT from the list of available ports and hit the "Connect" button. If successful you will see the COM port displayed in the status bar of the main PDT3 screen.

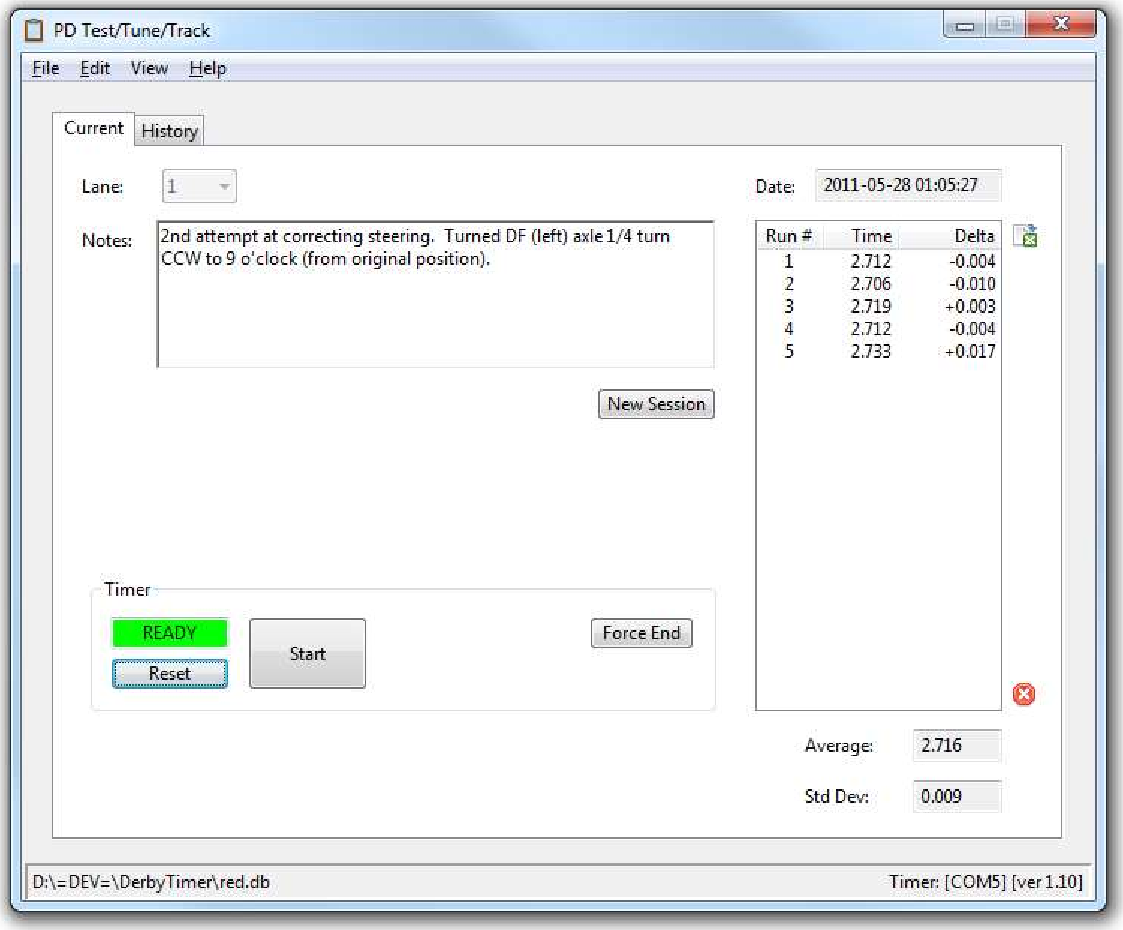
> ⚠ The timer needs to be in the 'READY' or 'FINISHED' state when starting PDT3 or manually connecting. If the timer is in the 'RACING' state PDT3 will not be able to connect.

The last step to perform before first use is to create a new database file by selecting "New" from the "File" menu. Typically, each car would have its own database file. The current database file is listed in the status bar of the main screen.

Current Session Tab

The main screen of the PDT3 utility, the Current Session tab, is shown below. This screen displays and records what is considered the current working session. As runs are performed the run times are added to the run list at the right. The overall average and standard deviation are automatically calculated as runs are added. In addition, the delta time difference between each run time and the overall average is calculated for each run.

The Lane selection list specifies which time is captured on a multi-lane track. Whether this value can be changed after the first run is performed is controlled by a flag in the "Option..." dialog.



Lane: 1

Date: 2011-05-28 01:05:27

Notes: 2nd attempt at correcting steering. Turned DF (left) axle 1/4 turn CCW to 9 o'clock (from original position).

Run #	Time	Delta
1	2.712	-0.004
2	2.706	-0.010
3	2.719	+0.003
4	2.712	-0.004
5	2.733	+0.017

Average: 2.716

Std Dev: 0.009

D:\=DEV=\DerbyTimer\red.db Timer: [COM5] [ver 1.10]

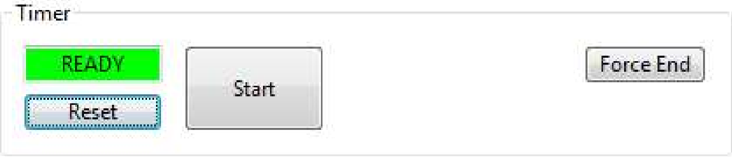
The "Notes" field is a free-form field available for a description of the session.

Erroneous run times can be deleted by selecting the run time in question and hitting the delete button to the lower right. The entire list of run times can be copied to the clipboard (in csv format) by hitting the copy button to the upper right.

The date/time displayed above the run time list reflects when the session was created. The "New Session" button will archive the current session data and start a new one.

Track / Timer Control

The Track/Timer control (shown below) on the "Current Session" tab is used to control the PDT. It will only be active if there is a successful connection to the PDT.



The "Reset" button sends a reset message to the PDT. The status field indicates the current status of the track/timer. Description of timer/track status:

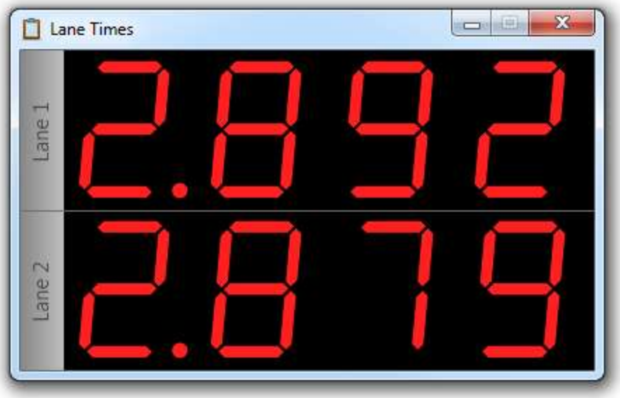
```
Not Ready - the track/timer is not ready (need to reset)
Gate Open - the start gate is open (cannot reset until closed)
Ready     - the timer/track is ready (to start)
Racing    - the race is in progress
Finished  - the race is completed
```

The "Start" button will send a start message to the PDT. If the PDT is equipped with the optional start gate solenoid circuit this message will open the start gate. If the PDT does not have that option simply initiate the run by manually opening the track's start gate – the software will automatically determine that the run has been started.

The "Force End" button will send a force end message to the PDT which will immediately send all lane times to the PDT3 software. This feature is typically used when for some reason (a derailment, for example) the car does not finish the run.

Lane Times Window

The "Lane Times" window, shown here, can be opened by selecting "Lane Times" from the "View" menu. This window displays the run times for each lane at the completion of the run.

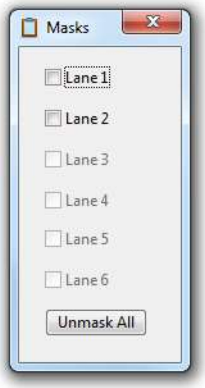


Lane Times

Lane 1 2.892

Lane 2 2.879

Lane Masks Window



Masks

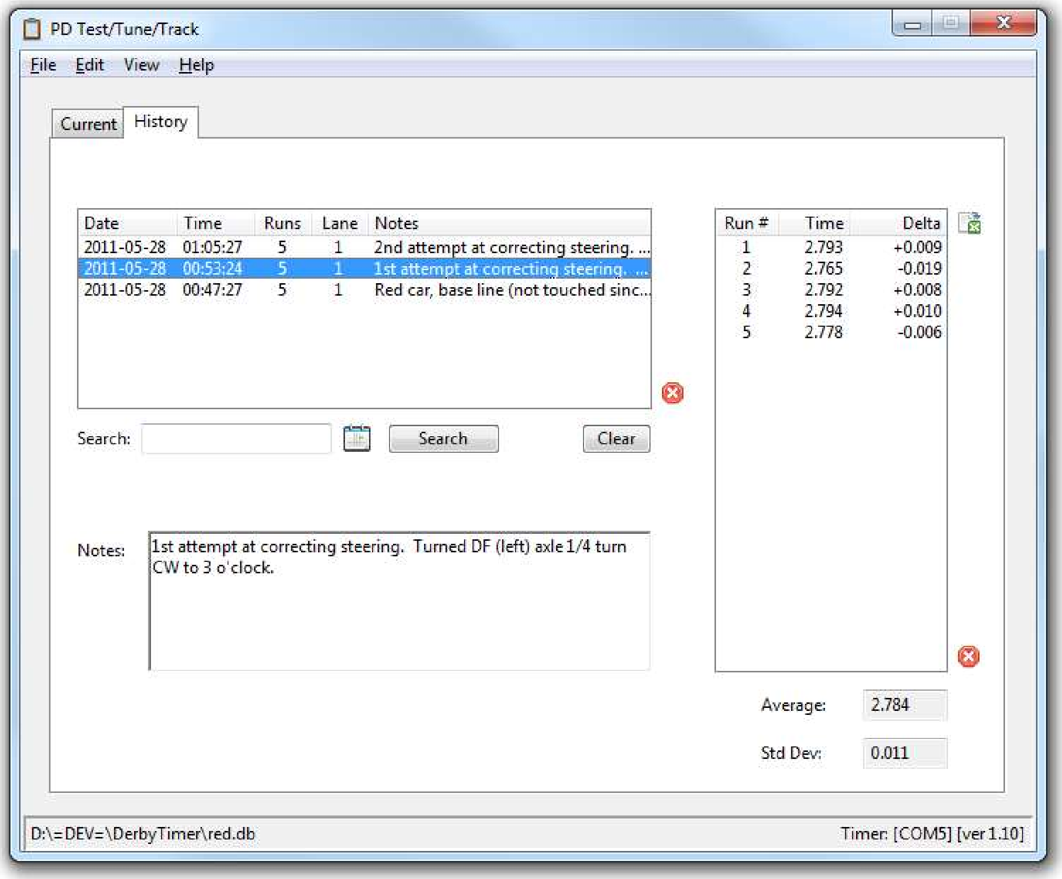
□ Lane 1
□ Lane 2
□ Lane 3
□ Lane 4
□ Lane 5
□ Lane 6

The "Lane Masks" window, shown here, can be opened by selecting "Lane Masks" from the "View" menu. On a multi-lane track this window allows you to mask unused lane(s). The "Unmask All" button resets (clears) all lane masks.

Masking a lane tells the PDT to not wait for a car to finish on an unused lane (since there is no car on that lane or lanes).

History Tab

The "History" tab, shown below, allows you to search for, browse and display all historical sessions. Data cannot be altered but sessions or runs can be deleted.



Date	Time	Runs	Lane	Notes
2011-05-28	01:05:27	5	1	2nd attempt at correcting steering. ...
2011-05-28	00:53:24	5	1	1st attempt at correcting steering. ...
2011-05-28	00:47:27	5	1	Red car, base line (not touched sinc...

Search:

Notes: 1st attempt at correcting steering. Turned DF (left) axle 1/4 turn CW to 3 o'clock.

Run #	Time	Delta
1	2.793	+0.009
2	2.765	-0.019
3	2.792	+0.008
4	2.794	+0.010
5	2.778	-0.006

Average: 2.784

Std Dev: 0.011

D:\=DEV=\DerbyTimer\red.db

Timer: [COM5] [ver 1.10]

BACK TO TOP

Created: 25 Feb 2015
Last updated: 6 May 2022