BILL V'S
DERBY TIMERS

Arduino Based Drag Race Timer for RC Scale Model Cars Version 4c

USER MANUAL

1 INTRODUCTION

The drag race timer discussed herein has been specifically designed for 1/10 scale Radio Controlled (RC) cars. This design has been tailored to mimic the features of an actual drag strip timing system including a functional light tree (aka Christmas tree), start line sensors that support vehicle staging and reaction time measurement, trap and finish line sensors to determine the vehicle speed and elapsed time. This design is intended to be a portable, easy to set up standalone system controlled by a single hand-held pushbutton. An optional race management software package is provided that runs on a PC that can control the race timer, display and save race results and allow for printing of time slips at the conclusion of each race. The software provided with this version has been tailored for a 1/10 scale quarter-mile (132 Ft) track with selectable speed indications in Miles-per-hour (MPH) or in Kilometers-per-hour (km/h). Figure 1 below provides a conceptual design of the timer control panel.

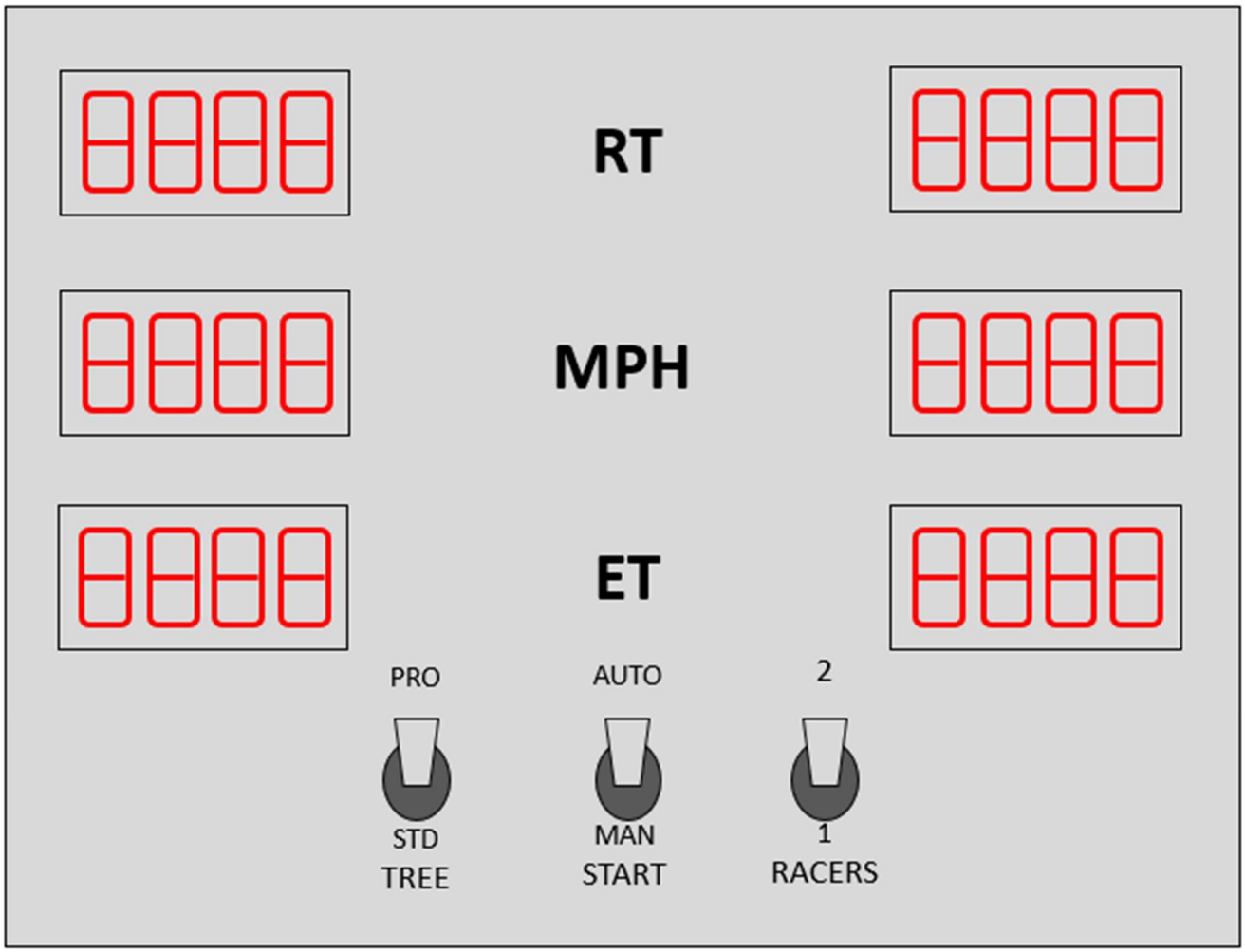


Figure 1 - Control Panel Conceptual Design

The Arduino UNO™ micro-controller board used in this design provides the interfaces to the optical lane sensors, light tree, numeric displays, mode select and race start switches, and executes the software that drives the Light tree lights, monitors the track sensors, performs the timing and displays the results on the numeric displays. Optionally, it sends the race results and track status to the race management software running on a PC (if used) for display and post-race processing.

The timer design supports three sets of infrared optical sensors positioned down the track as follows:

1. **Stage & Guard Sensors**: Positioned at the start line
    a. Support vehicle staging
    b. Measure reaction time (Green light to when vehicle leaves start line)
    c. Determines if a vehicle leaves the start line too early (i.e. red-lighting).
2. **Trap Sensors**: Work in conjunction with the finish line sensors to determine the vehicle speed at the end of the run. Positioned either 5.28 feet or 2 meters before the finish line depending on whether the timer is setup for speed measurement in MPH or km/h.
3. **Finish Line Sensors:** Positioned at the finish line. Measure vehicle race finish time.

Note: Many RC cars do not have adequate chassis/ground clearance to permit start line staging based on front wheel position. Hence, this version uses the car's body to detect its position for stage indications. A "guard" beam positioned about 2 inches forward of the stage beam is used to detect when the car leaves the start line.

A hand-held pushbutton tethered on a long cable is used to reset the timer after each race and manually start the countdown sequence (if in manual mode). The timer front panel (ref. Fig. 1) has six 4-digit LED numeric displays that display the reaction time, speed and elapsed time for each lane. Additionally, the control panel has three toggle switches providing the following selections:

1. Light tree Std./Pro mode operation
2. Manual/Auto start timing
3. 1 / 2 racers

The optional race management software running on the PC (see Fig. 9) provides display of the race start reaction time, trap speed, elapsed time, overall time and racer's win, draw, foul status. Race times are displayed down to three decimal places (0.001 seconds). The trap speed is calculated from the time difference between the trap sensor and finish line sensor. A selection made during the timer software setup allows the speed to be displayed in MPH or km/h. The race results can also be printed on a thermal receipt printer to provide a hardcopy record (Time Slip) to the race participants. All race results are saved to a comma-delimited data file that can be read by MS Excel or other spreadsheet software for post-race processing. The results are saved at the end of each race to ensure no data is lost. The race management software also provides the ability to view and print prior race results. Mouse clickable buttons allow the user to (1) ready the timer for the next race, (2) start the race, and as previously mentioned, (3) view race results from previously run races. A message box at the bottom of the race management screen provides for the display of timer and track status messages. The race management software is JAVA based and was developed using a free software package called "Processing" which is downloadable from the WEB at the Processing.org website (https://processing.org).

2 GETTING STARTED

2.1 Timer Components

The drag race timer consists of the components listed in Table 1 below.

Table 1. Timer Components

Component	Description / Function
Infrared Lane Sensors each consisting of 1 infrared transmitter module and 1 infrared receiver module. (Qty=8 - 4 per lane) Note that the transmitter modules are separate units powered by a 12 volt battery pack to provide for portability and ease of setup.	The infrared sensors sense and report when its invisible infrared beam is interrupted (broken) by an object. They are positioned along the track as follows: • Start line stage sensor – Positioned at the start line. Used for staging. • Start line guard sensor - Positioned just forward of the start line. Used to start elapsed timer, measure reaction time and determine foul start. • Trap line – Positioned either 5.28 feet or 2 meters before the finish line depending on whether the timer is setup for speed measurement in MPH or km/h. • Finish line – Positioned at the finish line. Used to measure total elapsed time.
Main Timer Unit	Contains all the electronics including the Arduino microcontroller for: • Monitoring the lane sensors, mode select switches and the hand-held pushbutton • Driving the Light tree • Performing the timing and timing calculations • Sending race results to the PC.
Hand-Held Race Start Button	Provides a means to start the race from a location other than the timer unit. Typically, this is done by the same person who is assisting with staging the cars at the start line.
LED Light Tree	Houses the staging, countdown sequence and disqualification lights.
Optional: PC (Laptop)	Computer that runs the race management display software.
Optional: Time Slip Printer	Thermal (receipt) printer that prints the time slips at the end of each race.

2.2 Component Setup

It is assumed the end user has already measured and laid out a 2-lane track. Figure 2 provides a typical setup of the timer components. The Light tree should be set up at a distance and height easily visible by participants at the start line.

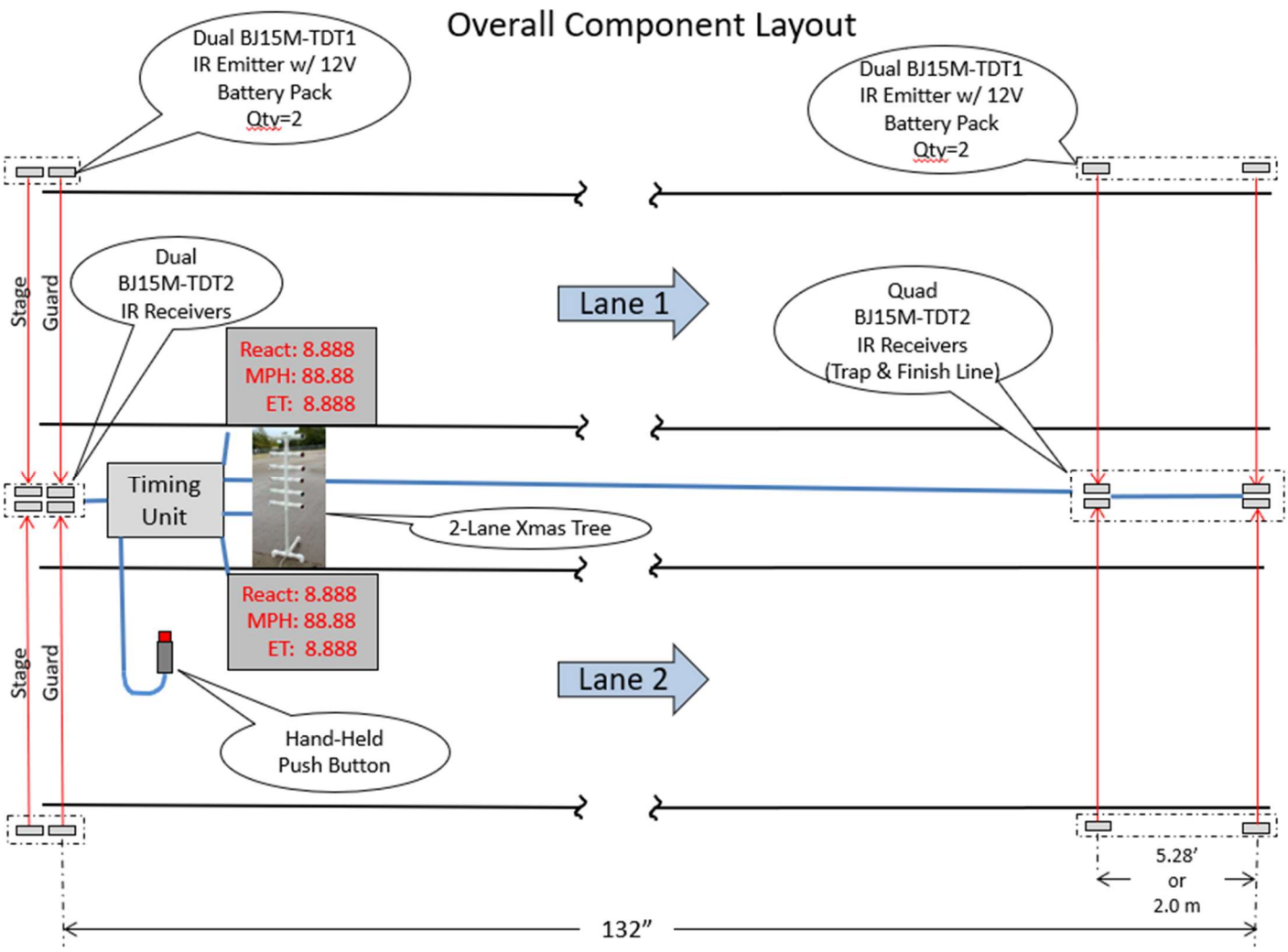


Figure 2 – Timer Component Layout

It is suggested that each trap and finish line emitter pair and receiver pair be mounted to a single board such that the sensor separation is fixed at either 5.28 feet (5' 3 3/8") or at 2.0 meters depending on whether the timer is setup for speed measurement is MPH or km/h respectively.

2.3 Timer Software Installation

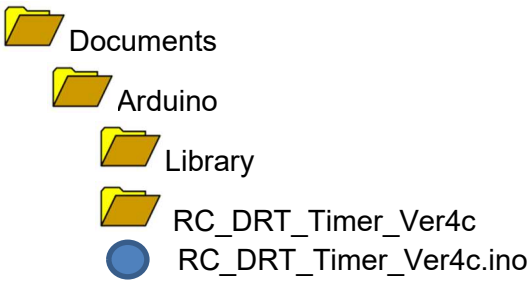
All required software except for the Arduino Integrated Development Environment (IDE) and the Processing Integrated Development Environment (IDE) are provided. Also, it is assumed the end user will acquire, install and setup the drivers required for the Time Slip printer, if used.

2.3.1 Arduino Integrated Development Environment Software Installation

With connection to the internet established, go to https://www.arduino.cc/ and follow the instructions to download and install the Arduino Integrated Development Environment (IDE). The WEB site also provides tutorials to help you every step of the way.

Once you have the Arduino IDE installed, create a subfolder called 'RC_DRT_Timer_Ver4c' under the "Arduino" folder (same folder containing the Arduino Library folder). The "Arduino" folder was created during the install, usually under your Documents folder. Now copy the file 'RC_DRT_Timer_Ver4c.ino' from the downloaded zip file to the 'RC_DRT_Timer_Ver4c' folder you just created.

This is a typical folder/file hierarchy but may vary:



NOTE: Installation of the Arduino Integrated Development Environment (IDE) also installs the necessary USB drivers for your PC to communicate with the Arduino via a serial RS-232 type (i.e. COM1, COM2, etc.) communication link. These drivers are also used by the race management software discussed in Section 2.3.1.

2.3.1.1 Library Installation

This version of the Arduino software uses the following libraries that must also be resident on your PC:

- SPI.h
- Wire.h (May already be installed by default)
- Adafruit_LEDBackpack.h
- Adafruit_GFX.h

To install these libraries, perform the following steps:

1. On the Arduino IDE, select: Sketch/Include Library/Manage Libraries to open the "Library Manager" window
2. On the "Library Manager" window search for and select the desired library, then click on the 'Install' button. Note that some libraries may already be installed by default.
3. Repeat step 2 for each library to install.

2.3.1.2 Uploading the timer code to the Arduino

Perform the following steps to upload the DRT timer code to the timer unit's Arduino microcontroller.

Step	Action	Comments
1	Ensure your PC is powered up and ready.	N/A
2	Apply power to the timer unit. **- - Note - -** Track Timer should always be powered up before connecting it to your PC.	N/A
3	Connect the track timer (Arduino board) to your PC USB port via the USB cable.	N/A
4	Double click on the 'DRT_Timer_Ver4c.ino' file to launch the file and bring up the Arduino integrated development environment.	Arduino integrated development environment (IDE) window is displayed with the DRT_Timer_Ver4c source code listing.
5	**MPH Or km/h Speed Setup** To set the timer for speed measurement in km/h, perform the following. Otherwise, skip this step. a. In the Arduino editor, scroll down to the portion of source code as shown in Figure 3. b. Modify the text at line 170 to read: *boolean kmh_Flag = true;* c. Select CTRL + S to permanently save your changes.	Arduino source code modified to calculate speed in km/h.
6	**Display Brightness Setup** To change the display brightness, perform the following. Otherwise, skip this step. a. In the Arduino editor, scroll down to the portion of source code as shown in Figure 3. b. Modify the text at line 171 to read: *int dispBrightness = n;* (where n is a value of 0 to 15 with 15 being the brightest). c. Select CTRL + S to permanently save your changes.	Arduino source code modified to set display brightness level.
7	Select the 'Tools/Board' pull-down menu to select/verify the "Arduino UNO" board is selected.	N/A
8	Select the 'Tools/Port' pull-down menu to select/verify the COM port selection (i.e. COM1, COM2, etc.).	Arduino COM port is selected / verified.
9	Select "→" (upload) to start the compile and upload process.	The program will compile and automatically upload to the Arduino board.

```
169   //USER SELECTABLE SETTINGS
170   boolean kmh_Flag = false;          //Set to true to display speed in km/h
171   int dispBrightness = 10;           //Range 0 - 15 (where: 0=dim, 15=max brightness)
```

Figure 3 – Timer Source Code User Selectable Settings

Refer to the Arduino website (https://www.arduino.cc/) for more in depth instructions and guidance if problems are encountered. Note that once the software has successfully been uploaded to the Arduino board, it is there permanently, unless overwritten by another upload. Hence, you only have to perform these steps once. Future use of the Arduino timer during your race events does not require an upload.

3 RUNNING A RACE

3.1 Setup

It is assumed the user has chosen a location that has a flat surface and is of sufficient length and width to accommodate the drag strip. Per the 2022 Radio Controlled Drag Racing League (RCDRL) rule book “the minimum width of each lane shall be 8 feet with 12 foot maximum. The length of the racing surface shall be exactly 132 feet. The racing surface may be asphalt or concrete or like surface.” Note that if the surface has a crown (curved), shims may be required to ensure the sensor beams are at the correct height.

3.1.1 Layout the Track

Tools required: Chalk line, Tape measure
Perform the following (Refer to Figure 2):

- Establish the location for the start line and using the chalk line, snap the start line onto the track surface. Ensure it is perpendicular to the direction of the track.
- Layout the lane 1 and lane 2 left and right boundaries. Provide enough separation between the two lanes for the timer unit and light tree. Lane width should be between 8 and 12 feet.
- Using the tape measure, measure a down-track distance of exactly 132 feet and using the chalk line, snap the finish line onto the track surface ensuring it is perpendicular to the direction of the track.
- Using the tape measure, measure a distance of 5.28 feet (5’ 3 3/8”) or 2.0 meters if setup is for speed measurements in km/h, back from the finish line and using the chalk line, snap the trap line onto the track surface ensuring it is perpendicular to the direction of the track.

3.1.2 Timer Equipment Setup

First Time Only Setup:

- On each IR receiver unit (BJ15M-TDT2) ensure the sensitivity adjustment screw and the D/L adjustment screw is set to the full clockwise position. Refer to the Autonics™ BJ Series datasheet for more information.

Perform the following (Refer to Figure 2):

- Position the timer and Light tree unit between the two lanes.
- Position the start line/stage sensor unit between the two lanes at the start line and connect its cable (W1) to J1 of the timer unit.
- Position the trap/finish line sensor unit at the finish line ensuring the sensors line up with the corresponding chalk lines. Extend the trap/finish line sensor cable between the two lanes and back to the timer unit and connect its cable (W2) to J2 of the timer unit.
- Position the battery powered IR transmitter units outside the lanes at the appropriate start, trap and finish line locations. Ensure they lined up with and point to their corresponding receiver unit on the opposite side of the lane.
- Connect the hand-held pushbutton cable (W4) to J4 of the timer unit.
- Connect the 12 volt power source to J5 of the timer unit.
- (Optional) If using a PC to run the race manager software, connect it to the Arduino USB port.

3.1.3 Timer Equipment Power up

Perform the following:

- Apply power to the timer unit. At startup the timer will perform a lamp test where the Light tree lights are cycled from top to bottom followed by the numbers 1.111 through 9.999 on the six 4-digit timer displays.
- Apply power to the battery powered IR transmitter units for both lane 1 and lane 2. Verify that the emitter unit’s green power indicator is illuminated. If not, check the batteries.
- Verify the green power/stability indicator and the red operation (beam detect) indicator on the IR receiver units are illuminated. If the green power indicator is illuminated and the red operation (beam detect) indicator is not illuminated check the sensor and/or emitter alignment since this is an indication that the receiver unit is not seeing the beam from the emitter unit.

- At each IR receiver unit obstruct the beam from the emitter unit and verify the red operation (beam detect) indicator extinguishes. Note, it may be required to adjust the receiver sensitivity. Refer to the Autonics™ BJ Series datasheet for more details.
- Momentary depress and release the hand-held pushbutton to issue a reset to the timer. Verify that both green Light tree lamps flash twice. If the lane 1 and/or lane 2 red foul lamps flash it means that one or more of the corresponding lanes IR receiver units is not detecting the beam. You must troubleshoot and correct the problem by rerunning the previous two steps.
- Install the protective covers, if any, over the IR emitter and receiver units.

3.2 Operation Overview

Operation of the Arduino based timer is straight forward. The timer code cycles through six states. They are:

- Track Not Ready
- Ready
- Staged
- Countdown
- Racing
- Finished

The “Track Not Ready” state is entered when the timer is commanded to ready for the next race and the software running on the Arduino senses that one or more of the optical lane sensors is obstructed or has failed. Either condition will cause the red foul light of the offending lane to blink twice. A Timer Reset must be performed by momentarily depressing the hand-held pushbutton once the problem is corrected. Correction of the problem will be indicated by two rapid blinks of the green Light tree lights.

The “Ready” state is entered when the timer is commanded to ready for the next race and the optical lane sensors are in the correct state to commence the staging process. As the cars move forward to the start line the stage lights will illuminate accordingly. When both cars are staged (both white stage lights illuminated) the timer advances to the “Staged” state. If a car advances to far forward such that the guard beam is broken, the red foul light will illuminate and the timer will return to the Ready state.

The “Staged” state is entered when both vehicles are staged signifying the timer is ready to start the race. If the timer is in the auto-start mode the timer automatically enters and starts the countdown sequence after a short random delay. If the timer is in the manual-start mode it will wait for a manual start command (pushbutton pressed) before entering the Countdown state to begin the countdown sequence. A manual start is performed by pressing the hand-held remote start pushbutton or by clicking on “Start Race” on the PC’s race management screen. Note that a car will not be automatically disqualified if it advances too far and breaks the guard beam. A car is permitted to re-stage without foul as long as the Countdown sequence has not yet begun.

The “Countdown” state is entered when the timer has sensed activation of the hand-held start switch or the Start Race button on the PC’s race management screen. If Auto-Start is enabled the timer will enter the countdown state automatically after a short random delay from when both cars were staged. During the countdown state the timer cycles the amber and green Light tree lights and monitors the start line guard sensors. If a guard sensor is tripped before the green light illuminates the corresponding red foul/disqualification light will illuminate and remain illuminated for the entire race.

The “Racing” state is entered when the green Light tree lights illuminate. At this point the start time is recorded and all lane sensors are monitored. When a sensor is tripped, the time at which it tripped is recorded. Note that

no race results are displayed until the race is complete. Vehicles have 10 seconds to cross the finish line or the timer will time out.

The “Finished” state is displayed when both cars have crossed the finish line or when the 10 second timeout duration has elapsed, whichever occurs first. The timeout is for cases where a car fails to cross the finish line or a lane was not used. In those cases, dashes will be displayed in the corresponding time displays for those sensors that were not tripped. A reset command (Hand-held push button pressed) must be issued to the Arduino timer to exit the ‘Finished’ state and ready it for the next race.

3.3 Operating the Timer – Stand Alone (No PC Connected)

It is assumed the user has properly set up and powered up the timer per section 3.1.

Perform the following steps to operate the timer in stand-alone mode (no PC connected):

Step	Action	Expected Results / Comments
1	At the timer’s control panel set the timer’s 3 mode select switches to the desired position. • TREE → STD / PRO • START → MAN / AUTO • RACERS → 1 / 2	Mode select switches set accordingly
2	Momentarily depress & release the hand-held pushbutton.	• Both green Light tree lights flash twice indicating the timer is ready for vehicle staging. • If the lane 1 or lane 2 red foul indicators flash instead, the timer has detected an obstructed or out of alignment lane sensor for the corresponding lane. Troubleshoot and correct the fault before proceeding.
3a	**1 RACER MODE**: If the 1 racer mode is selected only lane 1 is active and the vehicle must be staged on lane 1.	• As the racer’s vehicle breaks the staging beam the lane 1 stage light illuminates.
3b	**2 RACER MODE**: If the 2 racer mode is selected both lanes are active and a vehicle must be staged on each lane.	• As each racer’s vehicle breaks the staging beam the corresponding stage light illuminates. • Both vehicles must be staged in order to start the race.
4a	**AUTO START MODE** If the timer is in the Auto Start mode the timer will start automatically after a short random delay from when the vehicles are staged.	• The Light tree lights will begin their countdown sequence. If a racer trips the guard sensor before the green light illuminates, the corresponding red foul light will illuminate and remain illuminated until the timer is reset for the next race.
4b	**MANUAL START MODE** Momentarily depress and release the hand-held start button. ***Note**:The hand-held start button must be pressed and held for a minimum of 1/2 second to initiate a start.*	• The Light tree lights will begin their countdown sequence. If a racer trips the guard sensor before the green light illuminates, the corresponding red foul light will illuminate and remain illuminated until the timer is reset for the next race.

Step	Action	Expected Results / Comments
5	Allow the race to complete.	At completion of the race: • The 4-digit LED numeric displays will display the race results. • The green Light tree light of the winning lane will flash on/off for 15 seconds.
6	Record race results as desired.	N/A
7	Repeat steps 2 thru 6 for the next race.	N/A

4 USING THE RACE MANAGEMENT SOFTWARE (OPTIONAL)

***Note**: This section can be skipped if you don't plan on using the PC based race management software.*

Use of the race management software is optional. A PC (laptop) and USB cable are required. The software must be installed on your PC.

4.1 Race Management Software Installation

With connection to the internet established, go to https://processing.org and follow the instructions to download and install the Processing4 Integrated Development Environment (IDE). The WEB site also provides tutorials and step-by-step instructions to help you every step of the way. ***NOTE:** At the time of this writing the latest release of the Processing software was at Version 4.1.1.*

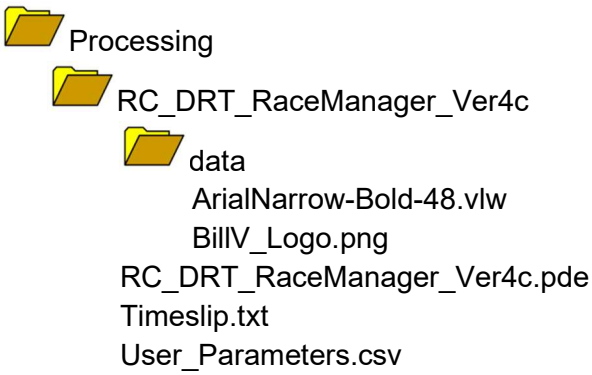
For Windows machines:

- Use File Explorer to view the contents of the Processing zip file (e.g. processing-4.1.1-windows-x64.zip) you downloaded from the processing.org website.
- Drag the unzipped processing folder into your C:\Program Files\ folder.
- Double click 'processing.exe' to launch the program and cause it to install. If everything goes right it should create a Processing folder under your Document folder and start with the Processing IDE screen displayed.
- Exit the Processing program.

Once you have the Processing4 IDE installed, perform the following:

1. Create a subfolder called 'RC_DRT_RaceManager_Ver4c' under the "Processing" folder.
2. Copy the files 'RC_DRT_RaceManager_Ver4c.pde', 'Timeslip.txt' and 'User_Parameters.csv' from the downloaded zip file to the 'RC_DRT_RaceManager_Ver4c' folder just created.
3. Now create a subfolder called 'data' under the 'RC_DRT_RaceManager_Ver4c' folder.
4. Copy the files 'ArialNarrow-Bold-48.vlw' and 'BillV_Logo.png' from the downloaded zip file into the 'data' folder.
5. Update the file path per Section 4.1.3 to match your PC's folder hierarchy (**IMPORTANT**).
6. Review Section 4.1.2 and make any desired changes to the User Modifiable Parameters.

The folder/file hierarchy should look like this:



4.1.1 Time Slip Printer Setup

NOTE: Use of a thermal receipt printer is optional. The time slip will print to your standard printer if you wish to use it instead as long as it is set as the default printer.

It is assumed the end user will provide the Time Slip printer and install/setup its drivers. During the driver setup the user must set the Time Slip printer as the **Default** printer. Additionally it may be required that a custom paper size be created/selected to avoid excessive amounts of paper to form feed at the end of each print out.

Other settings, depending on the make & model of printer may also be required including default font size and margins. This can be done in the printer properties menus. Note that when a time slip is generated, it is saved to a text file (Timeslip.txt) that can be manually printed using Notepad or other word processing software. This file is overwritten each time a new time slip is generated.

4.1.2 User Modifiable Parameters

The race management software has a handful of settings that can be modified by the user as follows:

- Display Title: Text displayed at the top of the screen
- Time Slip Header Line 1: 1'st line of user modifiable text printed at the top of the Time Slip
- Time Slip Header Line 2: 2'nd line of user modifiable text printed at the top of the Time Slip
- Time Slip Footer Line 1: 1'st line of user modifiable text printed at the bottom of the Time Slip
- Time Slip Footer Line 2: 2'nd line of user modifiable text printed at the bottom of the Time Slip
- Auto Print Copies: Number of Time Slip copies to automatically print at completion of each race

A separate file titled "User_Parameters.csv" has been provided in which user parameters are saved for future use. The file resides in the same folder as the race manager software (See Section 4.1). Modification of these parameters can be done in two ways. The first and easiest method is at program startup on the User Modifiable Parameters page (See Section 4.3). The second method is to use your favorite spreadsheet software to modify the parameter values directly as shown in Figure 4 below.

	A	B
1	PARAMETER	VALUE
2	Disp_Title:	Organization Name Goes Here
3	TS_Header1:	DFW Winternationals
4	TS_Header2:	RC Car Drag Racing
5	TS_Footer1:	Thanks for racing with us
6	TS_Footer2:	Come Back Soon
7	Auto print copies:	0

Figure 4 - Screenshot of User Modifiable Parameters

DEFAULT PARAMETER VALUES

A default set of the user modifiable parameters are stored in the program source code in case the parameters file is missing at program startup. The default set also includes a file path parameter that must be modified in the source code to match your computers file path setup (See Section 4.1.3). Figure 5 shows the section of source code that contains these default parameters.

```
32
33 /* * * * * * * * * * * * * * * * * * * * USER DEFINED PARAMETERS * * * * * * * * * * * * * * * * * * * * */
34 boolean kmh_Flag = false;                          //Set to true when speed from timer unit is in km/h
35 String Disp_Title = "Organization Name Goes Here"; //Text string to hold display title
36 String TS_Header1 = "DFW Winternationals";         //Time Slip header line 1 (33 Char Max)
37 String TS_Header2 = "RC Car Drag Racing";          //Time Slip header line 2 (33 Char Max)
38 String TS_Footer1 = "Thanks for racing with us";   //Time Slip footer line 1 (33 Char Max)
39 String TS_Footer2 = " ";                          //Time Slip footer line 2 (33 Char Max)
40 int Autoprintcopies = 0;                          //Number of copies to automatically print after race
41 //Set the file path for printing the Time Slip file
42 String filePath = "C:/Users/Bill/Timers/RC_DRT_Timers/RC_DRT_Ver4c-StandAlone/RC_DRT_RaceManager_Ver4c/";
43 /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
```

Figure 5 - Screenshot of User Defined Parameters Code

4.1.3 Modifying the File Folder Directory Path

NOTE: This step is required as part of the software installation process.

The race management software source file (RC_DRT_RaceManager_Ver4c.pde) has a string variable that defines the directory file path in which the Parameter and Time Slip files are saved. **Program limitations require that the file be saved in the same folder containing the program source code (sketch).** The file path must be updated to match the folder in which the files are saved on your computer. To update the file path, perform the following:

- In the Processing Display Environment scroll down to the "USER DEFINED PARAMETERS" section of the code (See Fig. 5) and find the line starting with: `String filePath = "C:/Users/….";`
- Edit the text inside the quotes with the correct file path for your PC.
- Select save (Ctrl S) to permanently save your changes.

4.1.4 Modifying the MPH / km/h Speed Setting

The speed display setting in either MPH or km/h is automatically set when the software is initialized and establishes communication with the timer unit. When communication with the timer unit is established, the timer unit will tell the race management software whether it is setup to measure speed in MPH or in km/h. The default setting is set to MPH. Perform the following to change the default setting km/h:

- In the Processing Display Environment scroll down to the "USER DEFINED PARAMETERS" section of the code (See Fig. 5) and find the line starting with: `Boolean kmh_Flag = false;…`
- Edit the text by replacing the word `false` with `true`.
- Select save (Ctrl S) to permanently save your changes.

4.2 Race Management Software Features

This version of the race management software provides the following features.

- COM port detection & selection at startup
- User editable Time Slip headers and footers for race event customization
- Ability to load a previously saved race event file and continue from where it left off
- Ability to view results from prior races
- Ability to enter a racer's ID number
- Manual and automatic Time Slip print capability
- Display of elapsed time (in seconds) while the race is in progress

4.3 User Modifiable Parameters Page

When the race manager program is started the User Modifiable Parameters screen is displayed from which you can select the COM port and optionally modify the user parameters as shown in Figure 6. The Speed Units setting is automatically updated when communication with the timer unit is established during Com Port selection. At that time the timer unit will tell the race management software whether it is setup to measure speed in MPH or in km/h and set the display accordingly.

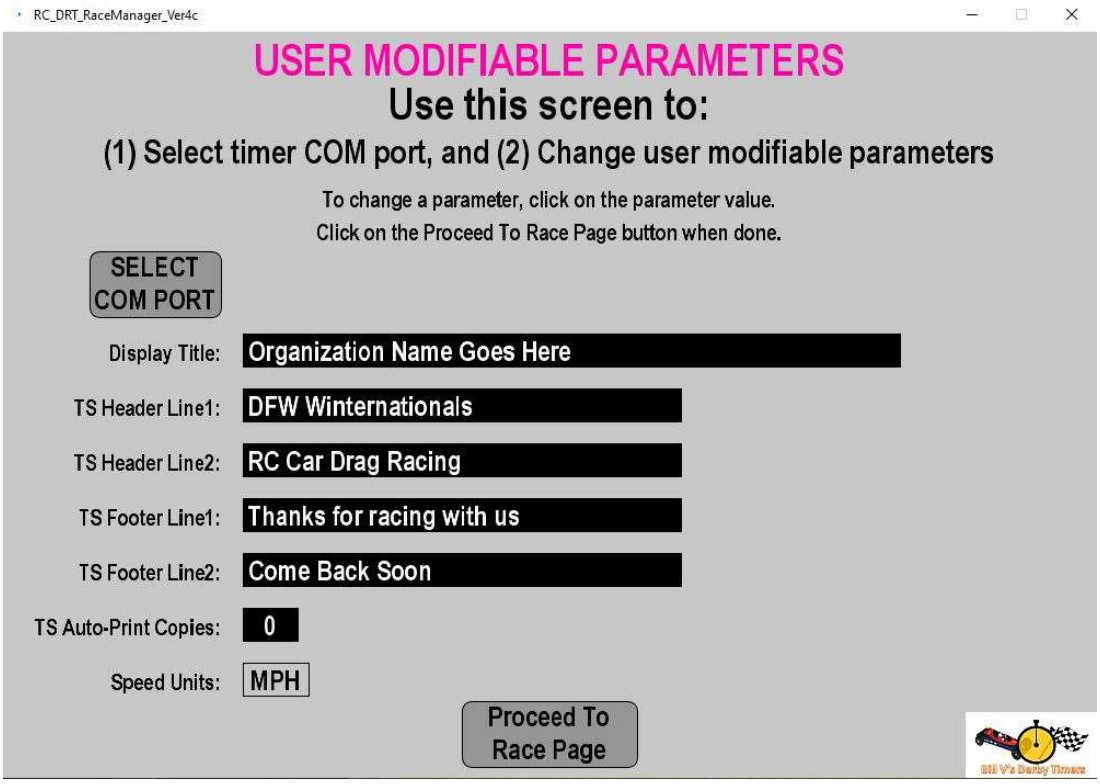


Figure 6 – User Modifiable Parameters Screen

4.3.1 COM Port Detection & Selection

To select the COM port to which the Arduino timer is connected, mouse click the SELECT COM PORT button and follow the pop-up dialogs. A COM port check is performed resulting in one of the following conditions.

- No COM ports Detected: Displays a warning pop-up message informing the user no COM ports were detected and then exits the program when the user clicks on OK.
- One COM port Detected: Assumes it is the Arduino COM port, automatically selects it and displays a pop-up message informing the user what COM port was selected.
- Two or more COM ports Detected: Displays a pop-up providing a list of available COM ports and allows the user to select which COM port to use. Once selected, a second pop-up message is displayed informing the user what COM port was selected.
- Selected COM Port unavailable: Displays a pop-up informing the user a COM port is not available and may be in use by another program.

If successful, a Com check OK message will be displayed to the right of the button. Note that you will not be allowed to proceed to the race page unless the COM port selection is successful.

4.3.2 Modifying the Display Title (Optional)

The title shown at the top of the race manager display can be modified as follows:

- On the User Modifiable Parameters screen mouse click anywhere in the Display Title text box. A dialog box will pop up in which you can type a new title.

- Select OK when done.

4.3.3 Modifying the Time Slip Header and Footer Text (Optional)

The time slip header and footer text can be modified as follows:

- On the User Modifiable Parameters screen mouse click anywhere in one of the time slip header or footer text boxes. A dialog box will pop up in which you can type a new header or footer line.
- Select OK when done.

Figure 7 shows an example time slip with 2 lines of header text and two lines of footer text.

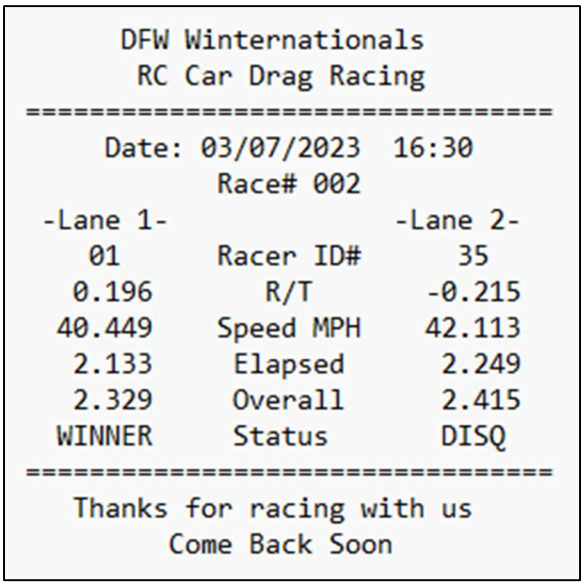


Figure 7 – Sample Time Slip

4.3.4 Modifying the Number of Time Slip Copies Automatically Printed

The race management software source file has a variable that defines the number of Time Slip copies to be automatically printed at the completion of each race. The software is delivered with this variable set to 0 (zero). To change the number of copies to be automatically printed, perform the following:

- On the User Modifiable Parameters screen mouse click anywhere in the TS Auto-Print Copies text box. A dialog box will pop up in which you can type a value in the range of 0 to 9. Typically, a value of 2 is selected (one for each racer).
- Select OK when done.

NOTE: It is recommended that this setting be kept at zero during initial startup until correct time slip printing has been verified using the “Print Time Slip” button.

4.3.5 Continuation from a Previously Saved Race Event

When the Proceed To Race Page button on the User Modifiable Parameters page is selected the user will be prompted if he wishes to continue from a previously saved race event (See Fig. 8). If he answers in the affirmative the user will be presented with a file selection popup from which to select the race event file (See Fig. 9). The filename will have the following format: “Race_Results_YYYYMMDDHHMM.csv”. These files can be found in the same folder containing the program code (Ref. Section 4.1.1). The selected file is then loaded and the results of the last race ran in the loaded file is displayed. The user can then continue the race event with the next available race number. Additionally, the user can also use the up/down arrows to view and print time slips of those races.

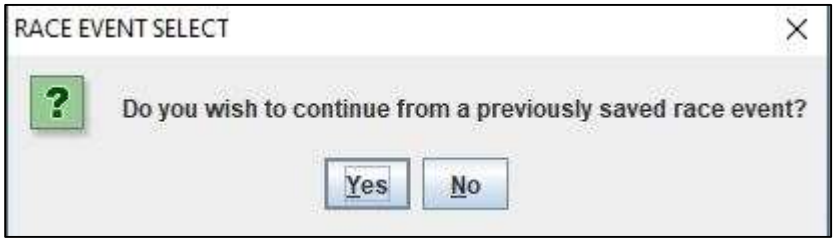


Figure 8 – Race Event Select Pop-up at Program Startup

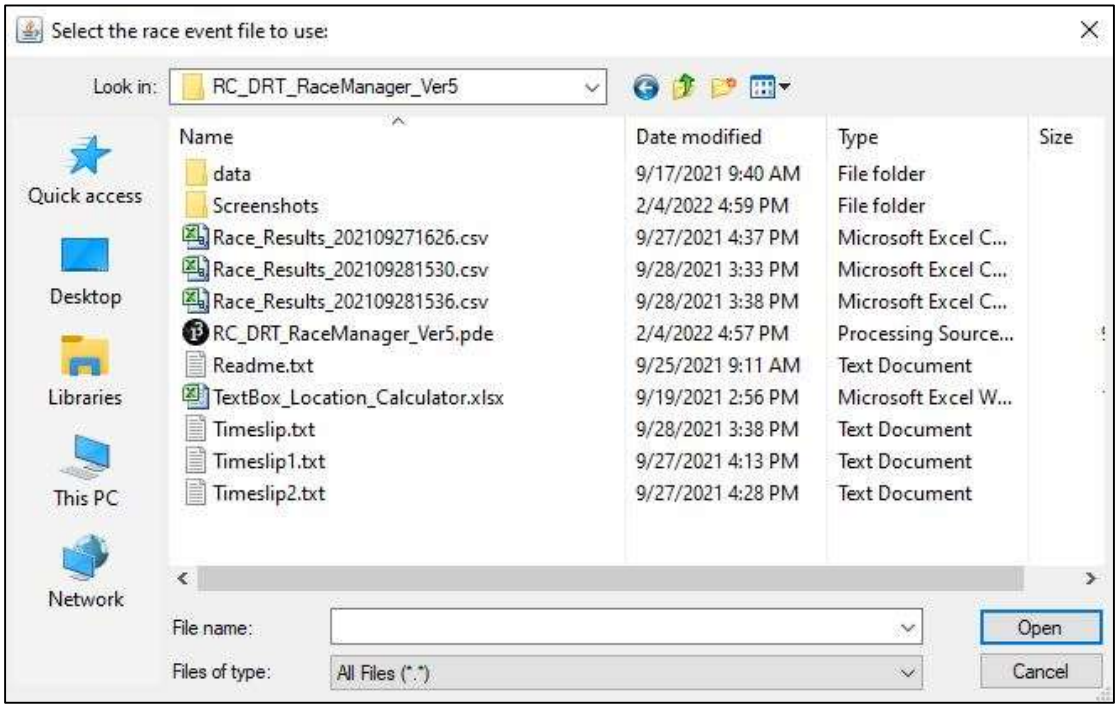


Figure 9 – Race Event File Selection Pop-up Example

4.3.6 Viewing Prior Ran Race Results

Data from each race ran, or loaded from a prior race event as discussed in the Section 4.3.5 above, is stored in a data table and can be easily recalled for viewing and printing of Time Slips. To view a prior ran race, click on the up/down arrow buttons in the Race # box. Note that the number cannot be advanced beyond the number of races in the data table. Note that the timer is disabled from starting a race while viewing previously run race results. Click on the up arrow until the new race number is again displayed to enable race start.

4.4 Race Management Display

The race management display (See Fig. 10a & 10b) displays the race results consisting of the reaction time, trap speed, elapsed time, overall time and racer's win/lose/disqualification status.

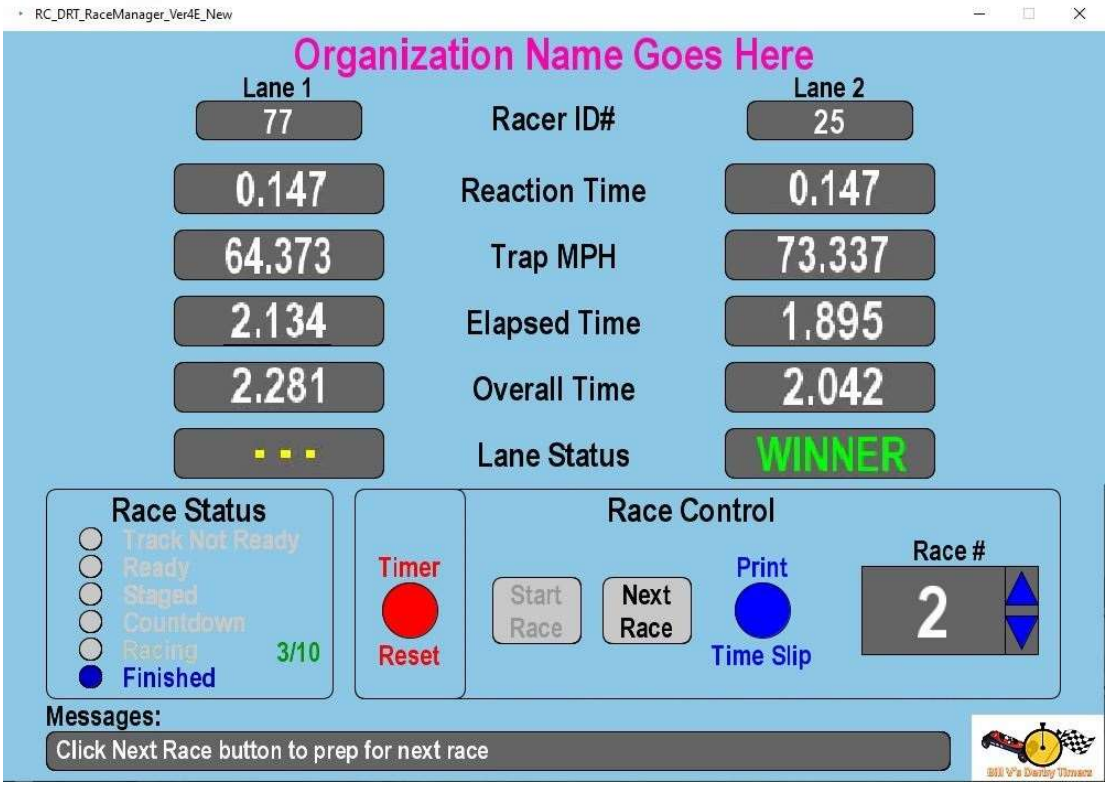


Figure 10a – Race Management Display (Speed in MPH)

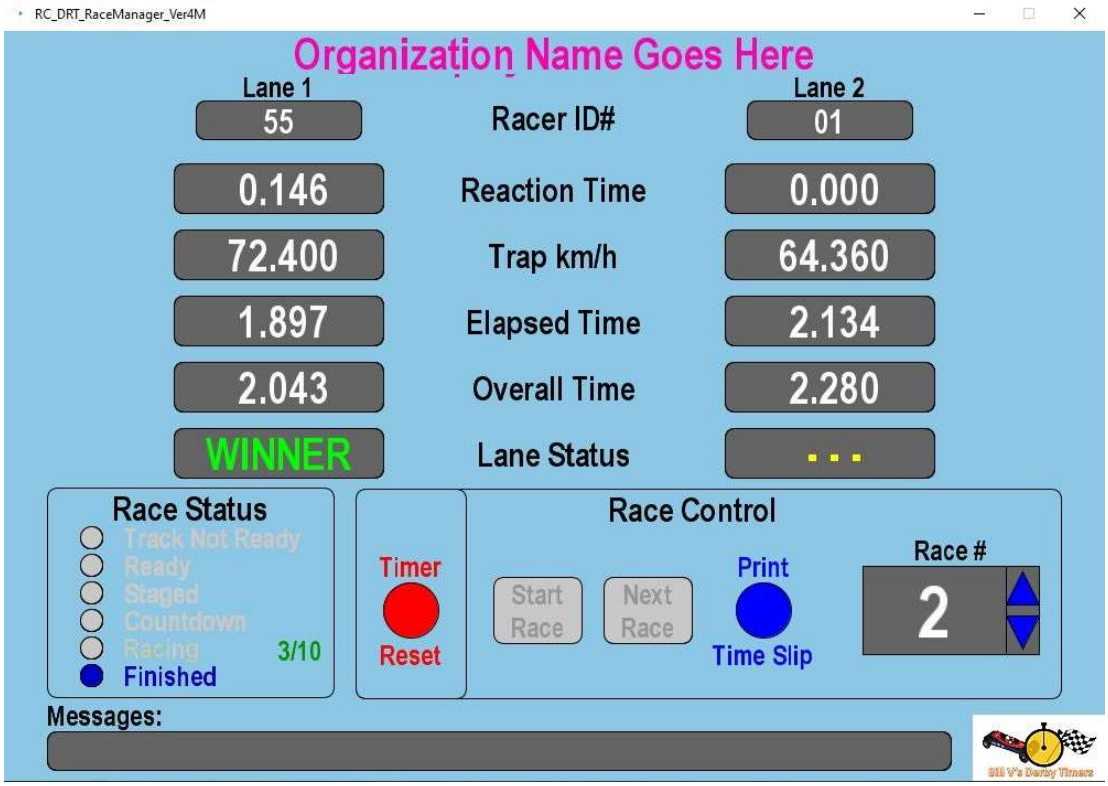


Figure 10b – Race Management Display (Speed in km/h)

4.4.1 Reaction Time

Reaction time is the time from when the race timer starts (green Light tree light illuminates) to when the racer's car moves far enough forward to trip the guard sensor. Time is measured and displayed to a millisecond (0.001 seconds). If the guard was tripped prior to illumination of the green Light tree light (racer red-lighted), the time is displayed as a negative value in red.

4.4.2 Trap Speed

The trap speed is calculated from the difference in the trap time and elapsed time (finish line sensor). Depending on whether the timer is setup to measure speed in MPH or in km/h the speed calculation is either S = 3600 / (Finish Time – Trap Time) or S = 7200 / (Finish Time – Trap Time) where S is the speed in Miles-MPH or km/h and the times are in milliseconds. Speed is displayed to 3 places to the right of the decimal point. If the racer fails to trip the trap or finish line sensor dashes will be displayed for the trap speed. Note that the track sensor separation between the trap and the finish line sensors is either set to 5.28 feet (1/1000 mile) for speed measurement in MPH or to 2.0 meters (2/1000 km) for speed measurement in km/h in order to simplify the floating-point calculations done in the software.

4.4.3 Elapsed Time

The elapsed time is the time from when the racer trips the guard sensor to when he trips the finish line sensor. Time is measured and displayed to a millisecond (0.001 seconds). If the racer fails to trip the finish sensor dashes will be displayed.

4.4.4 Overall Time

Typically, overall time is the time from when the green Light tree light illuminates (race timer starts) to when the racer crosses the finish line. In other words, it is the sum of reaction time plus elapsed time. However, if the racer leaves the start line early (before the green Light tree light illuminates), overall time is the same as the elapsed time.

4.4.5 Lane Status & Winner Determination

Lane status displays the status of each racer. Status is determined by the racer's overall time which is the sum of his reaction time and elapsed time and whether or not he fouled (disqualified) as a result of leaving the start line too early (aka red lighting). The timer software uses the rules laid out in the 2022 Radio Controlled Drag Racing League (RCDRL) rule book and applies the "first or worst" foul determination logic.

- DISQ – Is displayed if the racer "red lighted".
- WINNER – is displayed when that racer has the fastest time (reaction time + elapsed time).
- WINNER* - is displayed if the other racer has the fastest time but fouled (disqualified) making you the winner by default.
- DRAW – is displayed if both racers have identical finish times
- **- - -** (dashes) – are displayed if the racer loses or doesn't finish.

4.4.6 Race-In-Progress Timer Display

When a race is underway, a race-in-progress time is displayed in the Race Status box just to the right of the green "Racing" status indication (See Fig. 11). It displays the number of seconds that have elapsed (not to be confused with the racer's Elapsed time) since the race started. In addition, it also displays the race timeout duration just to the right of the "/" (forward slash) symbol. The example shown in Figure 8 shows the race has been underway for 3 seconds with a timeout duration of 10 seconds.

Figure 11 – Race-In-Progress Timer

4.4.7 Race Management Display Controls

The race management display has the following controls available to the user. Refer to Figure 9.

4.4.7.1 Next Race Button

The 'Next Race' button when clicked sends a reset command to the timer to ready it for the next race. In addition, it increments the race count to the next available race number which is then displayed in the Race # box. The 'Next Race' button is deactivated (greyed out) when the timer is in the "Staged", "Ready", "Countdown" and "Racing" state.

4.4.7.2 Start Race Button

The 'Start Race' button when clicked sends a start command to the Arduino timer to cause the Xmas tree to start its countdown sequence. The 'Start Race' button is deactivated (greyed out) whenever the timer is not in the 'Staged' state. This button mimics the functionality of the hand-held race start button.

4.4.7.3 Print Time Slip Button

The 'Print Time Slip' button when clicked causes a time slip of the currently displayed race results to be created and printed on the thermal receipt printer. Only one copy is printed each time the button is clicked. Used in conjunction with the 'Race #' Up/Down arrows, time slips can be printed for prior run races. Note that if the "Auto Print Copies" user setting is set to a value > 0 (See Section 2.3.4.4), one or more time slips will automatically be printed at the conclusion of each race.

4.4.7.4 Race # Up/Down Arrow Buttons

The Up/Down arrow buttons in the Race # box allow the user to select and view results from prior ran races. Used in conjunction with the 'Print Time Slip' button, this feature can be used to print time slips of prior run races.

4.4.7.5 Timer Reset Button

The 'Timer Reset' button provides a method to issue an independent RESET command to the Arduino timer. This is typically done to clear any lane sensor error messages reported by the timer unit.

4.4.7.6 Racer ID Entry

Racer ID numbers can be entered by the operator when the timer is in the Ready state. Entry of a racer's ID number is optional and not required to run the race. To enter a racer ID number, click on the desired Racer ID text box (highlighted in orange). A pop-up window will appear from which you can enter an alpha-numeric racer

ID number up to 6 characters in length. Refer to Figure 12 below. Note that the timer's remote hand-held pushbutton is disabled while the Racer ID pop-up window is active, thus preventing an inadvertent race start.

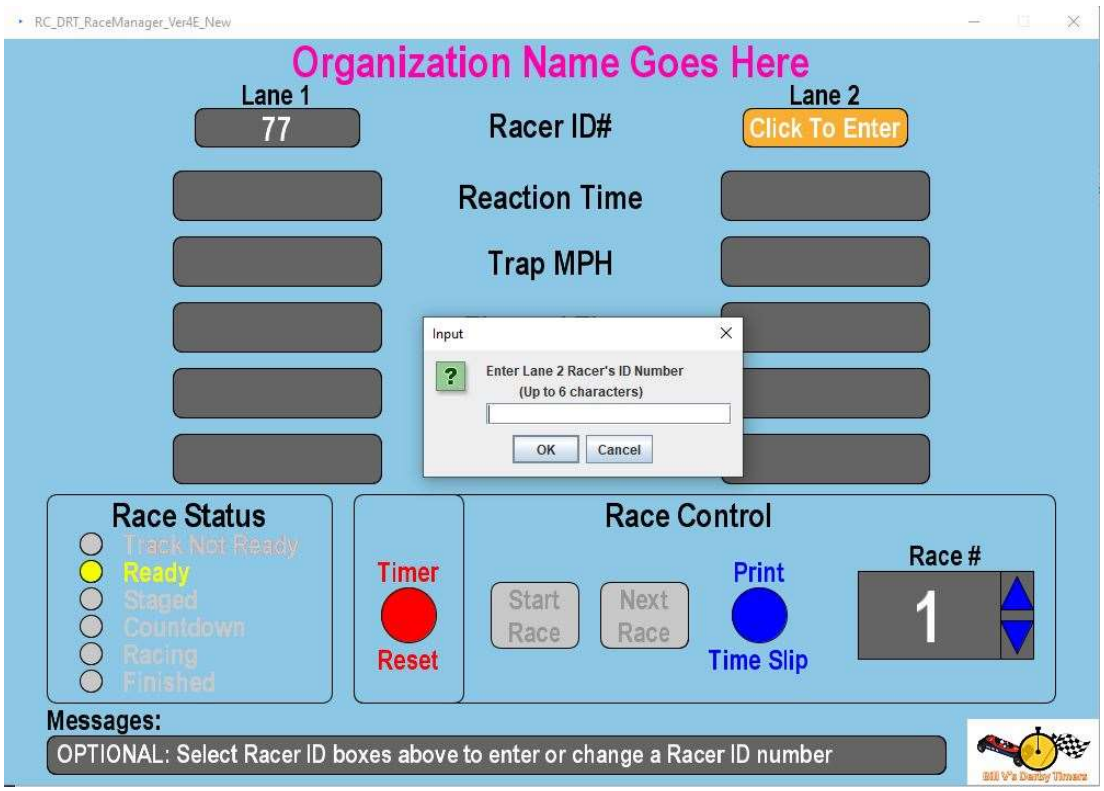


Figure 12 – Timer in The Ready State w/ Lane 2 Racer ID Entry Window Open

4.4.7.7 Light Tree Lamp Test

A self-test of the Light tree lamps can be initiated using the F1 function key. The lamp test is disabled when the timer is in the staged or racing state. **Note:** On many Windows PCs (typically laptops) it may also be necessary to simultaneously press the "Fn" key to bypass the Windows default function key action.

4.5 Running a Race Using the Race Management Software

This section describes the steps for running a race. Prior to running your first race ensure all timer components have been set up, powered up and tested per Section 3 of this document.

Step	Action	Expected Results / Comments
1	At the timer's control panel set the timer's 3 mode select switches to the desired position. • TREE → STD / PRO • START → MAN / AUTO • RACERS → 1 / 2	Mode select switches set accordingly
2	On the PC race management screen, click/select the Next Race button. Optionally, if a race has just been completed, the timer can be advanced to the next race by momentarily pressing/releasing the remote hand-held pushbutton.	• The next race number will be displayed in the RACE # box • Race Status changes to "Ready". • The Racer ID text boxes turn amber with text "Click To Enter" signifying the operator can enter Racer ID numbers (Optional). • If the Race Status does not change to "Ready" the timer has detected a fault and will display an error message at the bottom of the screen. In most cases the fault will be an obstructed or out of alignment lane sensor. Additionally, if an obstructed or out of alignment lane sensor is detected the red Xmas tree DQ light for the lane having the obstructed sensor will flash twice. The fault must be corrected before you can proceed.
3	(Optional) On the PC race management screen, click on the Race ID# box to enter a Racer ID number. Do for each racer.	A racer ID number up to 6 alpha-numeric characters is entered and displayed.
4a	**1 RACER MODE**: If the 1 racer mode is selected only lane 1 is active and the vehicle must be staged on lane 1.	• As the racer's vehicle breaks the staging beam the lane 1 stage lights illuminate. • Race Status on the PC changes from "Ready" to "Staged"
4b	**2 RACER MODE**: If the 2 racer mode is selected both lanes are active and a vehicle must be staged on each lane	• As each racer's vehicle breaks the staging beam the corresponding stage lights illuminate. • Both vehicles must be staged in order to start the race. • Race Status on the PC changes from "Ready" to "Staged" when both vehicles are staged.
5a	**AUTO START MODE** If the timer is in the Auto Start mode the timer will start automatically after a short random delay from when the vehicles are staged.	• The Light tree lights will begin their countdown sequence. If a racer trips the guard sensor before the green light illuminates, the corresponding red foul light will illuminate and remain illuminated until the timer is reset for the next race. • Race Status on the PC changes from "Staged" to "Countdown" and then to "Racing" when the green Light tree lights illuminate.

Step	Action	Expected Results / Comments
5b	**MANUAL START MODE** Start the race. This can be done via the hand-held start button or the Start Race button on the PC. ***Note**: If using the hand-held start button to start the race, it must be pressed and held for a minimum of 1 second to initiate a start.*	• The Light tree lights will begin their countdown sequence. If a racer trips the guard sensor before the green light illuminates, the corresponding red foul light will illuminate and remain illuminated until the timer is reset for the next race. • Race Status on the PC changes from "Staged" to "Countdown" and then to "Racing" when the green Light tree lights illuminate.
6	Allow the race to complete. This will occur when both racers have crossed the finish line or when 10 seconds has elapsed, whichever occurs first.	• The green Light tree light of the winning lane will flash on/off for 15 seconds • Race Status on the PC changes from "Racing" to "Finished". • Race results are displayed on the PC's race management screen. • If the "Auto Print Copies" setting is set to 1 or greater, the selected number of time slips is automatically printed.
7	Repeat Steps 1 through 6 for each race.	

4.6 Post-Race Processing

As mentioned earlier, all race results are saved to a comma-delimited (.csv) data file that can be read by MS Excel or other spreadsheet software for post-race processing. By default the file is saved in the same folder containing the program code (sketch). This can be changed by including the path in the filename variable (See the Processing.org website (https://processing.org) under the "saveTable" command). The filename for the data file saved to disk is created at program startup and includes a date/time value retrieved from the computer's clock. The filename structure is as follows: "Race_Results_YYYYMMDDHHMM.csv", where: YYYY = 4 digit year, MM = 2 digit month, DD = 2 digit day, HH = 2 digit hour and MM = 2 digit minutes. This guarantees a unique file each time the program is started and prevents any previous files from being accidentally overwritten.

Figure 12 shows a formatted example of the data saved to the race results file for each race. "L1" and "L2" stand for Lane 1 and Lane 2 respectively.

Race	L1 Racer	L1 Foul	L1 React	L1 Trap	L1 Speed	L1 ET	L1 Ovl	L1 Win
1	001	0	0.227	2.546	37.5	2.642	2.869	0
2	49	0	0.105	2.805	39.56	2.896	3.001	1
3	13	1	-0.07	2.872	38.298	2.966	2.966	0

L2 Racer	L2 Foul	L2 React	L2 Trap	L2 Speed	L2 ET	L2 Ovl	L1 Win	Date	Time
69	0	0.099	2.631	40.449	2.72	2.819	1	4/9/2021	15:35
023	0	0.291	2.78	37.895	2.875	3.166	0	4/9/2021	15:37
101	0	0.167	2.58	37.5	2.676	2.843	1	4/9/2021	15:39

Figure 12 – Example of Race Results File (formatted)

Race Results table legend:

- Race: Race Number
- L1 Foul: Lane 1 Foul (Disqualification) flag. 1 = Fouled 1'st, 2 = Fouled 2'nd
- L1 React: Lane 1 Reaction time
- L1 Trap: Lane 1 Trap time
- L1 ET: Lane 1 Elapsed time
- L1 Speed: Lane 1 Speed (in MPH or km/h)
- L1 Win Lane 1 Win flag: 1 = Winner
- L2 Foul: Lane 2 Foul (Disqualification) flag. 1 = Fouled 1'st, 2 = Fouled 2'nd
- L2 React: Lane 2 Reaction time
- L2 Trap: Lane 2 Trap time
- L2 ET: Lane 2 Elapsed time
- L2 Speed: Lane 2 Speed (in MPH)
- L2 Win Lane 2 Win flag: 1 = Winner
- Date: Date the race was ran
- Time: Time the race was ran

Appendix A
DIAGNOSTIC TEST PROCEDURE

This test procedure was written to assist in testing and troubleshooting the Arduino timer hardware and software. A minimum of two people using cell phones or walkie-talkies is recommended to perform this test.

Setup:

- Connect the PC to the main timer unit via the USB interconnect cable
- Bring up the Arduino Integrated Development Environment (IDE) on your PC
- Select and load the RC_DRT_Timer_Ver4c.ino file.
- On the Arduino IDE 'Tools' drop-down menu verify/select the correct COM port is selected.
- Note: You do not need to compile and upload the code onto your Arduino board unless you suspect it has been compromised.
- On the Arduino IDE open the serial monitor by clicking on the little magnifying glass near the upper right corner of the IDE display. Ensure the baud rate is set to 9600.

Perform the following test procedure.

Step	Action	Expected Results
1	At the timer's control panel set the timer's 3 mode select switches as follows • TREE → STD • START → MAN • RACERS → 2	Mode select switches set accordingly
2	Ensure the hand held Race Start Switch is not pressed and the lane sensors are properly illuminated	N/A
3	On the serial monitor enter the letter R in the command line, press ENTER or click on the Send button.	The two green Light tree lights flash on/off two times in rapid succession. The message "RDY" is displayed on the monitor. If you get the message "TRK <sensor ID>" followed by "NRD" one or more lane sensors as identified by their sensor ID is not ready. Troubleshoot and correct the problem before continuing.
4	Obstruct (block) the light illuminating the Lane 1 stage sensor.	The Lane 1 Light tree white stage lights illuminate.
5	While continuing to block the light illuminating the Lane 1 stage sensor, enter the letter R in the command line, press ENTER or click on the Send button.	The Lane 1 Light tree red foul light will flash twice indicating a lane 1 sensor is not ready. The message "TRK Stg1" followed by "NRD" is displayed indicating the stage sensor for Lane 1 is not ready.
6	Restore the light illuminating the Lane 1 stage sensor then enter the letter R in the command line, press ENTER or click on the Send button.	The Lane 1 Light tree white stage lights extinguish. The two green Light tree lights flash on/off two times. The message "RDY" is displayed on the monitor.

Step	Action	Expected Results
7	Repeat steps 4 through 6 for the Lane 2 stage sensor.	Results are the same as noted above except for Lane 2. The message “TRK Stg2” followed by “NRD” is displayed indicating the start line sensor for Lane 2 is not ready.
8	Obstruct (block) the light illuminating the Lane 1 guard sensor.	The Lane 1 Light tree red foul light will illuminate.
9	While continuing to block the light illuminating the Lane 1 guard sensor, enter the letter R in the command line, press ENTER or click on the Send button.	The Lane 1 Light tree red foul light will flash twice indicating a lane 1 sensor is not ready. The message “TRK Grd1” followed by “NRD” is displayed indicating the stage sensor for Lane 1 is not ready.
10	Restore the light illuminating the Lane 1 guard sensor then enter the letter R in the command line, press ENTER or click on the Send button.	The two green Light tree lights flash on/off two times. The message “RDY” is displayed on the monitor.
11	Repeat steps 8 through 10 for the Lane 2 guard sensor.	Results are the same as noted above except for Lane 2. The message “TRK Grd2” followed by “NRD” is displayed indicating the start line sensor for Lane 2 is not ready.
12	Obstruct (block) the light illuminating the Lane 1 trap sensor.	N/A
13	While continuing to block the light illuminating the Lane 1 trap sensor, enter the letter R in the command line, press ENTER or click on the Send button.	The Lane 1 Light tree red foul light will flash twice and then remain extinguished. The message “TRK Trp1” followed by “NRD” is displayed indicating the Lane 1 trap sensor is not ready.
14	Restore the light illuminating the Lane 1 trap sensor then enter the letter R in the command line, press ENTER or click on the Send button.	The two green Light tree lights flash on/off two times. The message “RDY” is displayed on the monitor.
15	Repeat steps 12 through 14 for the Lane 2 trap sensor.	Results are the same as noted above except for Lane 2. The message “TRK Trp2” followed by “NRD” is displayed indicating the trap sensor for Lane 2 is not ready.

Step	Action	Expected Results
16	Repeat steps 12 through 14 for the lane 1 and lane 2 finish line sensors.	Same result as steps 12 through 14 except the reported sensor ID will correspond to the sensor being obstructed. Also, for Lane 2 sensors the Lane 2 Light tree red foul light will flash twice. The message “TRK Fin1” or “TRK Fin2” followed by “NRD” is displayed indicating the corresponding finish line sensor is not ready. **NOTE**: If the reported sensor does not correspond to the sensor being obstructed, check the sensor cable connections to ensure they are plugged into the correct receptacle.
17	Remove all sensor obstructions and ensure the hand-held Start Switch is not depressed.	N/A
18	On the serial monitor enter the letter T in the command line, press ENTER or click on the Send button.	The Light tree lamp test sequence is executed where each set of lamps will cycle on/off at a 0.5 second rate followed by the numbers 1.111 thru 9.999 being displayed on the timer’s 4-digit numeric displays. The message “@” is displayed on the monitor when the test completes.
19	On the serial monitor enter the letter R in the command line, press ENTER or click on the Send button.	The message “RDY” is displayed on the monitor. No Light tree lamps are illuminated.
20	Obstruct and continue to obstruct the beam of both the Lane 1 and Lane 2 stage sensors.	The white stage lights are illuminated on the Light tree. The message “STG” is displayed on the monitor.
21	While continuing to obstruct the Lane 1 and Lane 2 stage sensors momentarily press & hold the hand-held Start Switch.	The message “CNT” is displayed on the monitor. The Light tree amber and green lights cycle through their countdown sequence at a 0.4 second rate. When the green lamps illuminate the message “RAC” is displayed on the monitor.
22	Wait about 10 seconds.	After 10 +/-1 seconds the green Xmas tree lights extinguish and the following messages are displayed on the monitor: “Lane1: 0 9999.999 9999.999 9999.999 0” “Lane2: 0 9999.999 9999.999 9999.999 0” Followed by the message “FIN”.
23	Remove the obstruction from the Lane 1 and Lane 2 stage sensors.	The white stage lights are extinguished on the Light tree.
24	On the serial monitor enter the letter R in the command line, press ENTER or click on the Send button.	The message “RDY” is displayed.
26	On the timer control panel set the TREE switch to the “PRO” mode.	N/A

Step	Action	Expected Results
27	Obstruct and continue to obstruct the beam of both the Lane 1 and Lane 2 stage sensors.	The white stage lights are illuminated on the Light tree. The message “STG” is displayed on the monitor.
28	While continuing to obstruct the Lane 1 and Lane 2 stage sensors enter the letter S in the command line, press ENTER or click on the Send button.	All three sets of amber Light tree lights illuminate simultaneously followed 0.4 seconds later by the green lights. The message “CNT” is displayed on the monitor. When the green lamps illuminate the message “RAC” is displayed on the monitor.
29	Allow the timer to time out.	After timeout the green Xmas tree lights extinguish and the following messages are displayed on the monitor: “Lane1: 0 9999.999 9999.999 9999.999 0” “Lane2: 0 9999.999 9999.999 9999.999 0” Followed by the message “FIN”.
30	Remove the obstruction from the Lane 1 and Lane 2 stage sensors.	The white stage lights are extinguished on the Light tree.
31	Momentarily press & release the hand-held Start Switch.	The message “RST” followed by the message “RDY” is displayed on the monitor.
32	On the serial monitor enter the letter D in the command line, press ENTER or click on the Send button. **Note**: If Speed Setting indicates “km/h”, the variable ‘kmh_Flag’ has been set to “true” in the Arduino source code.	The message “DEBUG RESULTS” is displayed followed by the following: Pro Mode: True Autostart: False One Racer: False Race State: 1 Speed Setting: mph
33	On the timer control panel set the TREE switch to the STD.	N/A
34	On the serial monitor enter the letter D in the command line, press ENTER or click on the Send button. **Note**: If Speed Setting indicates “km/h”, the variable ‘kmh_Flag’ has been set to “true” in the Arduino source code.	The message “DEBUG RESULTS” is displayed followed by the following: Pro Mode: False Autostart: False One Racer: False Race State: 1 Speed Setting: mph
35	On the timer control panel set the RACER switch to the 1.	N/A
36	On the serial monitor enter the letter D in the command line, press ENTER or click on the Send button. **Note**: If Speed Setting indicates “km/h”, the variable ‘kmh_Flag’ has been set to “true” in the Arduino source code.	The message “DEBUG RESULTS” is displayed followed by the following: Pro Mode: False Autostart: False One Racer: True Race State: 1 Speed Setting: mph
37	On the timer control panel set the TIMER switch to the AUTO.	N/A

Step	Action	Expected Results
38	On the serial monitor enter the letter D in the command line, press ENTER or click on the Send button. **Note**: If Speed Setting indicates “km/h”, the variable ‘kmh_Flag’ has been set to “true” in the Arduino source code.	The message “DEBUG RESULTS” is displayed followed by the following: Pro Mode: False Autostart: False One Racer: True Race State: 1 Speed Setting: mph
39	On the serial monitor enter the letter C in the command line, press ENTER or click on the Send button.	The message “@-mph” is displayed. If the message “@-kmh” is displayed, the variable ‘kmh_Flag’ has been set to “true” in the Arduino source code.
40	Restore the timer mode switches to their desired position. On the serial monitor enter the letter R in the command line, press ENTER or click on the Send button.	N/A The message “RDY” is displayed.
--	TEST COMPLETE	

Appendix B
Arduino Timer Serial Communication Details

Communication between the Arduino based timer and the PC is via the USB interface which has been set up as a serial link running at 9600 baud, 8 bits, no parity, and 1 stop bit (9600/8-N-1). ASCII character strings transmitted between the timer and the PC are used to control the timer as described in the table below.

Table B1 – Timer Serial Communication Commands

ASCII Command	Direction	Description
R	PC-to-Timer	Reset – Resets the Arduino timer
S	PC-to-Timer	Race Start – Commands the Arduino timer to start the race.
D	PC-to-Timer	Debug – Returns status of the 3 timer mode switches, current race state and whether the timer is set to measure speed in MPH or km/h. See steps 32 thru 38 of the diagnostic test procedure.
H	PC-to-Timer	Hold – Sends a hold command to the timer disabling the hand-held pushbutton from starting a race. Used when lamp test is being executed.
C	PC-to-Timer	Continue – Rescinds the hold command.
T	PC-to-Timer	Test - Commands the Arduino timer to execute the lamp test.
@	Timer-to-PC	Acknowledge character (for testing purposes only)
@-mph	Timer-to-PC	Com check acknowledge message to let the race manager software running on the PC know that the timer has been set to measure speed in MPH.
@-kmh	Timer-to-PC	Com check acknowledge message to let the race manager software running on the PC know that the timer has been set to measure speed in km/h.
NRD	Timer-to-PC	Not Ready – Informs the race management software the timer is in the ‘Not Ready’ state.
STG	Timer-to-PC	Stage - Informs the race management software the timer is in the ‘Staging’ state.
RDY	Timer-to-PC	Ready – Informs the race management software the timer is in the ‘Ready’ state (i.e. Both cars staged and ready to launch).
CNT	Timer-to-PC	Countdown – Informs the race management software the timer is in the ‘countdown sequence’ state.
RAC	Timer-to-PC	Racing – Informs the race management software the timer is in the ‘Racing’ state.
FIN	Timer-to-PC	Finished – Informs the race management software the timer is in the ‘Finished’ state.
RST	Timer-to-PC	Manual Reset – Informs the race management software that the hand-held switch issued a manual reset command.

ASCII Command	Direction	Description
TRK -or- TRK, x, y, …	Timer-to-PC	Track Status – Informs the race management software that the track is not ready for the next race because the Start Switch or one or more optical lane sensors are obstructed or out of alignment. If a lane sensor is obstructed the TRK message will include code for the effected sensors. Valid codes are: • Stg1 – Lane 1 Stage sensor • Stg2 – Lane 2 Stage sensor • Grd1 – Lane 1 Stage sensor • Grd2 – Lane 2 Stage sensor • Trp1 – Lane 1 Trap sensor • Trp2 – Lane 2 Trap sensor • Fin1 – Lane 1 Finish Line sensor • Fin2 – Lane 2 Finish Line sensor

In addition to the above serial command messages the Arduino timer passes the race results to the PC display software via two ASCII strings (one for each lane) having the following format:

Lane1: x aaaa.aaa bbbb.bbb cccc.ccc dddd.ddd y
Lane2: x aaaa.aaa bbbb.bbb cccc.ccc dddd.ddd y
Where:

- x = Lane foul (disqualification) status (0=OK, 1=Fouled 1'st, 2=Fouled 2'nd.)
- aaaa.aaa = Lane reaction time
- bbbb.bbb = Lane trap time
- cccc.ccc = Lane Finish time
- dddd.ddd = Lane speed (in MPH or km/h)
- y = Win flag where 1 = win, 0 = lose or draw

Note that the Arduino software adds 1000.000 seconds to the actual time when formatting the race results string to ensure floating point time values always have the same character position within the string. The added 1000.000 seconds is subtracted from the received results by the race management software.

Troubleshooting: Use of the Arduino IDE serial monitor or another serial terminal program can be used to observe these commands or to send the reset "R" command to the timer. Refer to the diagnostic test procedure in Appendix A to assist in troubleshooting any issues. Feel free to contact the author via email at billv923@outlook.com for additional help.