# M&J Timing and Scoring

Copyright 2014

by Murray Peterson

Contents

[M&J Timing and Scoring 1](#_Toc382564044)

[Description 2](#_Toc382564045)

[Installation 4](#_Toc382564046)

[Setup for first use 5](#_Toc382564047)

[Config file 5](#_Toc382564048)

[Driver data file 6](#_Toc382564049)

[Class data file 6](#_Toc382564050)

[Timing data files 6](#_Toc382564051)

[COM Ports 6](#_Toc382564052)

[Timing 7](#_Toc382564053)

[Manual timing 7](#_Toc382564054)

[Automatic timing 8](#_Toc382564055)

[Keyboard shortcuts 9](#_Toc382564056)

[Running an event 9](#_Toc382564057)

[Scoring 11](#_Toc382564058)

[MJannounce 13](#_Toc382564059)

[Fixing up scores 13](#_Toc382564060)

[Custom scoring 13](#_Toc382564061)

## Description

The M&J timing software suite is designed for simplicity of operation. Some notable features and differences from other timing software:

1. All data files are comma separated files (.csv) with a header line that notes the meaning and/or use of each column within the file. Extra columns are ignored by the applications, so they may be added by the user at will. The first column is always the unique key field and may not be moved; all other columns may be re-arranged as desired.
2. Event files are always named by using the current date in yyyy\_mm\_dd format (e.g. 2012\_09\_30), so the user never needs to name an event (or configure the software every time he starts an event).
3. There is only one driver data file (aka membership file), and a copy is automatically made for each event (for the drivers that change cars and/or class).
4. Rarely changed items (timer port numbers, paths to files) are configured in a config file, not within the various applications.
5. You may exit or kill the timing program at any time. The timing program saves data to the timing file every time there is a change.
6. Scoring is available in realtime (hit the F1 key to toggle between the timing screen and the scores).

Included are these programs:

1. MJtiming  
   The program that does the timing and sends times to a display (compatible with Axware hardware). If no timing hardware is available, times and penalties may be entered manually.
2. MJCmdlineScoring  
   A command line scoring program. This program can be run from batch scripts, which is a convenient mechanism for storing the settings used to score an event. It also is useful for saving event scores to a text file by redirecting the output.
3. MJOveralls  
   A command line scoring program that will calculate overall scores from all existing events in a given folder. It reads CSVData files that have been created by MJCmdlineScoring or MJtiming.

## Installation

There is no installation program for the M&J software. Simply unzip the files and place into a folder of your choice (suggested folder is c:\mjtiming). For easy access, you should create a shortcut to mjtiming.exe on your desktop.

Next, create more folders in the location of your choice:

1. Event data folder  
   This is where all event timing information will be stored
2. Folder for storing driver data and class data   
   Initial versions of these two files are included in the zip file (\_driverData.csv and \_classData.csv).

If you unzip the media into c:\mjtiming, then all folders and files will exist, and the config file will be correct with the exception of the timer and display COM ports.

## Setup for first use

### Config file

In the folder c:\mjtiming\config, there is a configuration file (ConfigData.csv) that must be edited before you can start using the system. You usually modify this file by starting mjtiming and selecting the configuration tab.

The first column is the name of the parameter (or setting). The second column is the value of the setting, and the third column is a description of the parameter (helps you edit the file without referring to this document). Parameter values that you must set:

* driverDataFile  
  Location of the driver data file (there is only one). This file is where all of the drivers are defined – their car number, name, car class, rookie status, membership status, and so on.
* ClassDataFile  
  Location of the car class definition file.
* EventDataFolder  
  Folder where all event timing data is stored.
* BackupDataFolder  
  Folder where timing data backup files are placed.
* TimerPort  
  COM port (e.g. COM3) for connecting to timing hardware. MJtiming accepts Axware, Race America and RaceBeam protocol strings (Adddddd for start time, Bdddddd for stop time).
* PrimaryDisplayPort  
  COM port for sending times to the primary display. Set to blank if no display is connected.  
  The protocol string is 0x80fffddd (time string is in reverse order). Works with Race America, Chronomix and RaceBeam displays.
* SecondaryDisplayPort  
  COM port for secondary display. Not currently implemented.
* ConesGetPAXed  
  A “Yes” means that the driver's PAX factor will be applied to his total time (including cone penalties). This is the SCCA standard. e.g.  
   PAX time = (raw time + (cones \* 2.0)) \* (PAX factor)  
  A “No” means that PAX will be applied to the driver's raw time and then the cone penalties will be added. e.g.  
   PAX time = (raw time \* PAX factor) + (cones \* 2.0)

If you unzip the supplied media into [c:\mjtiming](file:///C:\Users\cloudconvert\server\files\mjtiming) and then run mjtiming once, then all folders and files will exist, and the config file will be correct. You will only need to set the timer and display COM ports for your system on the timing computer.

### Driver data file

You will need to enter or update all driver data into this file before the start of every event. As the season progresses, this should be a simple job of verifying that the data is correct, and putting a “y” or “yes” into the registered column for each driver that is attending an event. The file provided includes a few drivers for examples.

As soon as you run MJtiming, a copy of the driver data file is placed into the event data folder. This is done because drivers can change car or class from event to event, so the correct driver data for that event is always available at a future date.

**Note: If you change the registration data during an event, only the new runs will show the modified data. Regular scoring and timing will be correct, but the program does not go back and edit past entries shown on the realtime display.**

### Class data file

The file provided contains the SCCA class definitions for the current year (not including ladies classes), as well as AWD, FWD, and RWD classes for winter rallycross events. You may add your own club's classes as desired.

This file will only need to be updated at the beginning of every year.

### Timing data files

Each day that an event is run, all timing data is stored in a single file in the event data folder. The file is automatically named *yyyy\_mm\_dd*\_timingData.csv. e.g. if you run an event on June 1, 2015 then the timing data is stored in a file named 2015\_06\_01\_timingData.csv

This automatic naming of timing data files appears to be a bit dictatorial, but it prevents timing personnel from naming events randomly or placing them in unknown locations.

### COM Ports

Select this tab to discover which COM ports are used by your timing hardware and display hardware. Start the application, plug in the timing hardware, and select the tab to refresh the COM port list. The port that shows up (e.g. COM3) is the one that should be entered into your ConfigData.csv file. Repeat the operation for discovering your display sign port (if it does not share the same port as the timing hardware).

Once your initial configuration is complete, test it by starting MJtiming; you should see no errors in the message text box.

## Timing

Once your config file is filled in correctly, start MJtiming. There should be no error messages in the log message window.

The fields that you can edit are:

1. Car number  
   Enter the car number and hit enter or the right arrow key or the down arrow key. If the car number is defined in the driver data file, then the driver's name, PAX value and car description will be displayed. If this data isn't displayed, then that car number is not defined in the driver data file – you need to edit the driver data file and enter the driver's information. This need not be done immediately, since this information is only used for scoring – the timing program really doesn't care if it is present or not.
2. Penalty  
   The allowed values for penalties are:
   1. Any number from 1 to 99  
      This is the number of cones struck or displaced by the driver
   2. DNF  
      The driver went off course or did not finish the run. You can enter the letter “d” or the letter “x” for shorthand.
   3. RRN  
      A driver was granted a rerun and this run is thus marked as invalid. You can enter the letter “r” for shorthand.
   4. The number 0  
      This will cause the penalty field to be blanked out (no penalty).
3. Time  
   If you are manually timing vehicles, then you can manually enter the run time. If you are using timing hardware, then this will be filled in automatically. You can still modify this field, so be careful to not edit the value with automatic timing.
4. Set  
   To allow two day scoring within a single calendar day, there is a selection for set 1 or set 2 or set 3 (fun runs). This merely marks each run as being made on set 1 or set 2 or fun runs. If you made an error, you can edit the set value for the runs that were in error.

### Manual timing

Manual timing is simple. For each run, enter the car number, the penalties and the run time in seconds (e.g. 42.015). If you have a display connected to the configured COM port, the time will be sent to the display.

**Note**: **You must always enter a time, even if the driver is marked with a DNF penalty.**

### Automatic timing

Once the timing hardware is set up, press the large “Timer Stopped” button; it should change colour and now read “Timer running”. If this doesn't happen, check your COM port settings in the config file and view the COM ports tab to verify that the COM port is accessible.

When a car trips the start beam, you will see the word “Running” appear in his time column. When he trips the finish beam, the word “Running” is replaced by his actual time.

* False beam trips  
  If a beam is tripped accidentally by a marshal (or by debris blocking the sensor), you can reset the trip by pressing the “Reset Start” or “Reset Finish” buttons.
* Missed beam trips  
  If a sensor failed to trip, you may manually trip it by pressing the “Trigger start” or “Trigger stop” buttons. Because this was a manually entered trigger, the car is automatically marked as getting a rerun.

Experiment with the four buttons before running your first event. You can delete the timing values for the last row by pressing “Reset stop” and then pressing “Reset start”. You can add a new time (999.999 + RRN) by pressing “Trigger start” and then pressing “Trigger stop”

## Keyboard shortcuts

* F1  
  Switches between the timing tab and the scoring tab. Useful for showing people their score when they come up to the timing station.
* F2  
  Switches between the timing tab and the registration tab. Useful for repairing errors or omissions. Note that existing timing entries will not be refreshed to reflect the new driver information, but the correct information will be used when scoring.
* Ctrl-L  
  Moves the cursor to the first empty row in the timing entry grid. Useful when you have scrolled back to look at something earlier in the day.
* F3  
  Reset last start trigger. Used to cancel a false start trigger.
* F4  
  Reset last stop trigger. Used to cancel a false stop trigger.
* F5  
  Create a start trigger. Used when a car started, but the start did not trigger. Automatically marks the car as having a rerun (RRN).
* F6  
  Create a stop trigger. Used if a car fails to trigger the stop beam. Automatically gives the driver a rerun (RRN) and sets his time to 999.99

## Running an event

1. Mark all drivers as not registered (there is a button that will do this for you in a single operation).
2. Register each driver  
   If the driver is new, press the “Add driver” button. He is automatically given a new number that is higher than the SeedCarNumber config value. If he requests a different number, enter it into the Number column and hit return. The number must be a **unique** car number (for this event), so you will get an error popup if the number is not unique. The popup will display the driver who already has that number and asks you if you wish to forcibly apply the number to the new driver. The previous owner of the number will have an “x” appended to the number.
3. Verify every driver's information (member, rookie, class, etc.)  
   Mark him as registered by putting a “y” or “Yes” in the registered column.  
   **Note:** *If you wish to add a new column, stop MJTiming, then open the driver data csv file in a spreadsheet application and add the new column header at the end. Then restart MJtiming****.*Note:** *To delete a driver, highlight the line by clicking on the left side of the line, and then press the delete key.*
4. Select proper class and group for driver. Groups are used to collect vehicles into larger groups for scoring and run heat purposes. For example, the group STOCK can be used to score all SCCA stock classes together (PAX is used within that group). If you want each individual class, then just put the group name equal to the class name.
5. Heat Assignments tab  
   This tab will display the counts for each class and class group, as well as make suggestions for how to split the drivers into two or three heats. This is the only software that pays any attention to the “Yes” in the *registered* column.
6. Run timing  
   Press the “Start Timing” button and verify that no errors occur on opening the COM port. For extra verification, have someone break the start and finish beams and verify that a time was recorded correctly. Use the reset stop and reset start buttons to delete the test time. Or easier, just mark that row with a RRN penalty.

**Note: Never try to delete an entry in the timing grid. Instead, just mark the entry with a RRN penalty, and the run will be ignored by the system.**

## Scoring

Scoring is available in real time (press the F1 key to switch between the scoring tab and the timing tab).

The leftmost checkbox area allows you to select which scoring “groups” will be displayed in the scores. Select each one that you desire:

* Summary  
  A quick summary showing information about each driver that may be of use to an announcer. This information is also accessible in a separate application called mjannounce.exe.
* Run Times  
  A list of each driver in numerical order, and a list of each one of his runs (and penalties)
* Raw Times  
  Overall ranking of drivers sorted by RAW time
* PAX Times  
  Overall ranking of drivers sorted by PAX time
* Class Times  
  Ranking of drivers within each unique group. If the groups are the same as the classes, then each class will be shown individually.
* Cone Counts  
  Drivers ranked by the number of cones they hit during the event.
* Team Scores  
  Ranking of each team (sum of the best run of each driver in the team). Teams are determined by the name in the Team column in the registration tab.  
  Teams can be scored only by raw times or only by PAX times. To select which one, place either RAW or PAX in the Group field for each driver (in the registration tab). If a team’s drivers are all given their regular group (e.g. STOCK), then the team will be scored for both raw and PAX.

The next checkbox allows you to select scoring variants on the above selections:

* Rookie times  
  If selected, then all of the scoring groups will be repeated, with rookie drivers only

The next checkbox allows you to select scoring based on set number indicated for each run:

* Set1 only  
  Display scores only for runs marked as Set1
* Set 2 only  
  Display scores only for runs marked as Set2
* Set 1 + Set 2  
  Score drivers based on their best run from Set 1 added to their best run from Set 2
* Best single run  
  Ignore the Set number and just assign scores based on a driver’s best run

The “# official runs” box allows you to select the maximum number of official runs for each day (defaults to 99). You can use this instead of (or in addition to) the “Fun” checkbox in the timing tab. Thus, if you select 3 official runs, then the 4th and subsequent runs for each day will be scored as a fun run.

When you wish to actually create a score file, enter a title string and press the “Create score files” button. The score file will be created in the eventdata folder, as well as a file called <yyyymmdd>\_scoreCMD.bat. The bat file is a script that will regenerate the event’s scores when you run it; this is useful for fixing up registration or classing errors after the event has finished. Just edit the driverData.csv file and then double click the scoreCMD.bat file.

Some clubs like to treat morning and afternoon as two completely separate events for purposes of overall scoring. To do this:

a) Enter the date in the day1 slot

b) Select set1-only and any other score stuff you want

c) Enter title

d) Press Create Scores button

e) Repeat a-d for set2-only

The program will have created these files:

<date>d1\_CSVData.csv

<date>d1\_\_scores.txt

<date>d2\_CSVData.csv

<date>d2\_\_scores.txt

Now, when you run the mjoveralls command, the d1 and d2 scores will be scored as separate events.

The important files for the overalls are the CSVData files. When you run mjoveralls.exe, he looks for every file in the folder named as <date>\*\_CSVData.csv.

### Scoring a specific group (Open PAX or OPAX)

Many clubs have an Open PAX or OPAX group with these features:

1. The people running in this group all run in the same heat, regardless of their class
2. The scores for this group are only derived from the first 3 runs

Do the following:

1. Create a group in the classes file and call it OPAX (or Open Pax) (if one isn’t already there)
2. Set the PAX value for this class to 3 (the number of runs that will be scored)
3. For each driver running in OPAX, set the registration “Group” field to OPAX

Any group defined in the classes file that has a PAX value greater than 2 will be treated specially; that PAX value will be used as a maximum number of runs for scoring purposes.

### Overall scoring

Whenever you score an event and hit the "Create score files" button in the scoring tab, several files get created in the eventdata folder. They are

* <date>\_\_scores.txt  
  Scores with options as selected in the scoring tab
* <date>\_\_scoreCMD.bat  
  This is a batch script to perform the same scoring action from the command line  
  Uses mjCmdlineScoring.exe instead of mjtiming.exe
* <date>\_CSVData.csv  
  This file is the rolled up summary of the scoring for this event.  
  Used by mjoveralls.exe to create season totals

The important file for overall scoring is <date>\_CSVData.csv

1. Copy all of the <date>\_CSVData.csv files to a folder of your choice (or leave them in the eventdata folder if you wish).
2. Open a cmd shell in that folder
3. Run a command much like this:   
   c:\mjtiming\bin\mjoveralls.exe -path . -norookie -title "VMSC 2014 Championship points" > \_overalls.txt
4. Better yet, there is a script in the config\_templates folder called \_score\_cmd\_overalls.bat. Copy this file into your season’s scoring folder and edit it to match your desired parameters, title, etc. Then, whenever you have a new event or correct an event, you need merely double sclink on the script to regenerate the season’s overall scores.

Mjoveralls takes a few arguments (use -h argument for a complete list)

* -best <n>  
  best <n> results (assuming you score <n> events out of <x> events)
* -rookiebest <n>  
  same, but for rookies (they have different requirements in many clubs)
* -norookie  
  pretty obvious
* -title "title for the season’s scores”  
  pretty obvious
* -classfile  
  path to class csv file (e.g. c:\mjtiming\config\\_classData.csv)  
  This is only needed if you are running this script when the mjtiming class file is different than the one used for this folder’s scores.

Mjoveralls looks for all files named <date>\_CSVData.csv and scores up the totals from those events.

If you look in mjtiming\config\_templates, you will see some sample batch files that can be used as templates for your overall scoring.

## MJannounce

Mjannounce is an application that can be run at the same time as mjtiming, so if the computer has an extended display, the announcer can see the summary information without requiring the timing operator to switch between timing and scoring tabs. Mjannounce will automatically detect changes to the data files and automatically update the display.

Mjannounce will also act as a webserver that will serve up live scoring results. To make it work, you need to do this:

1. Attach a router to the computer that is running mjannounce
2. Configure router to forward incoming requests to web server computer, port 80
3. Turn off firewall on web server computer (or configure firewall to allow incoming request)
4. Run Mjannounce as Administrator
5. Press the “Start webserver” button
6. Clients can access the web page from their browser by going to this address: http://192.168.0.100/ (assuming timing computer address as 192.168.0.100)
7. One trick for ease of use is to print a QR code on your course maps that translates to the URL of <http://192.168.1.100>. Racers can scan it with their smartphone and it does all the magic for them as long as they are connected to the results WiFi network.
8. If your router supports DNSmasq features, you can assign any (or all) URLs to be forwarded to the web server. Enable DNSmasq and configure it with “address=/#/192.168.2.100”. Note that many routers can be updated with dd-wrt firmware, which supports DNSmasq: <http://www.dd-wrt.com/site/index>

### Fixing up scores

When you wish to fix up scoring errors on the same day as the event, you can just fix up the data in the registration tab and regenerate the scores. However, if the date has changed (e.g. it’s the next day), you must fix up errors by directly editing the <date>\_driverData.csv file.

The reason for this is because people change cars and classes frequently, so a given date’s driver information is only considered valid for that day.

### Custom scoring

The scoring calculation module is an accessible assembly, so you can write your own scoring application if you wish (e.g. HTML output instead of text output). The documentation folder contains a small C# program called sample.cs that you can use as a starting guide.