

Contest Spreadsheets

Here are three LibreOffice spreadsheets for different contests:

- [RAC Canada Day / Winter](#) (Last updated on July 1, 2021)
- [Winter Field Day](#) (Last updated on Jan 28, 2023)
- [Field Day](#) (Last updated on Jan 28, 2023)

All three spreadsheets are used in very similar ways, but they are different to account for the different rules and ways to calculate the points.

Most of the introduction below uses RAC's contest as an example but the same ideas apply to the other contests.

RAC Canada Day / Winter Contests

The spreadsheet was last updated on November 29, 2020 to [Cabrillo V3.2](#) for the contest on Dec 18-19 2020.

Everything you need to know about the RAC Canada Day or Winter Day contest is located on the [RAC website](#). It can be a bit overwhelming to digest for first-time testers, so here's an overview.

Logs

During the contest, the following information must be logged for each station:

- Frequency in kHz (for example, 14.125 MHz should be entered as 14125)
- Mode (CW or PH)
- Date and Time in UTC (for example 2020-07-01 and 0135)
- Callsign
- Signal Report (for example 59 on phone or 599 on CW)
- Exchange (Two letter province code for Canadian Stations, or the serial number for other stations).

At the end of the contest, logs must be [submitted to RAC](#). They will accept paper logs for submissions with less than 100 entries, but they really prefer electronic logs, which must be submitted in the *Cabrillo* format.

A Cabrillo file is really just plain text file formatted in a very specific way. For example, here's a copy of my Cabrillo file from last summer (with only a few entries as example):

```
START-OF-LOG: 3.0
CREATED-BY: RAC_Contests.ods v2020.07.04 by VA7FI
CALLSIGN: VA7FI
LOCATION: BC
CONTEST: RAC CANADA DAY
CATEGORY-OPERATOR: SINGLE-OP
CATEGORY-BAND: ALL
CATEGORY-MODE: SSB
```

```
CATEGORY-POWER: LOW
CATEGORY-TRANSMITTER: UNLIMITED
CLAIMED-SCORE: 11564
CLUB: Sun Coast Amateur Radio Club Society
NAME: Patrick Truchon
ADDRESS: REDACTED
ADDRESS-CITY: Roberts Creek
ADDRESS-STATE-PROVINCE: British Columbia
ADDRESS-POSTALCODE: V0N 2W1
ADDRESS-COUNTRY: Canada
EMAIL: va7fi@rbox.me
OPERATORS: @VA7FI
SOAPBOX: First Canada Day contest on my own. Lots of fun!
QS0: 14186 PH 2020-07-01 0008 VA7FI          59 BC      VE3PJ          59 ON
QS0: 14198 PH 2020-07-01 0012 VA7FI          59 BC      WB0TEV         59 14
QS0:146520 PH 2020-07-01 0112 VA7FI          59 BC      VE7DX          59 BC
QS0: 14211 PH 2020-07-01 0139 VA7FI          59 BC      N800           59 201
QS0: 14152 PH 2020-07-01 0148 VA7FI          59 BC      VE2CJR         59 QC
QS0: 14165 PH 2020-07-01 0159 VA7FI          59 BC      VE6RAC         59 AB
END-OF-LOG:
```

The first 21 lines give RAC all the information they need about the operator. The other lines show the contacts made, ending with an END-OF-LOG tag. For more information about the Cabrillo format, see this [RAC pdf](#), or the [WWROF website](#).

Regular testers use logging programs which can generate these files while providing features that facilitate the logging process during the contest. Many people have recommended [N1MM](#) as being the best one. RAC also has its own Microsoft Windows [contest program](#) for its own contests. This is what the spreadsheet looks like:

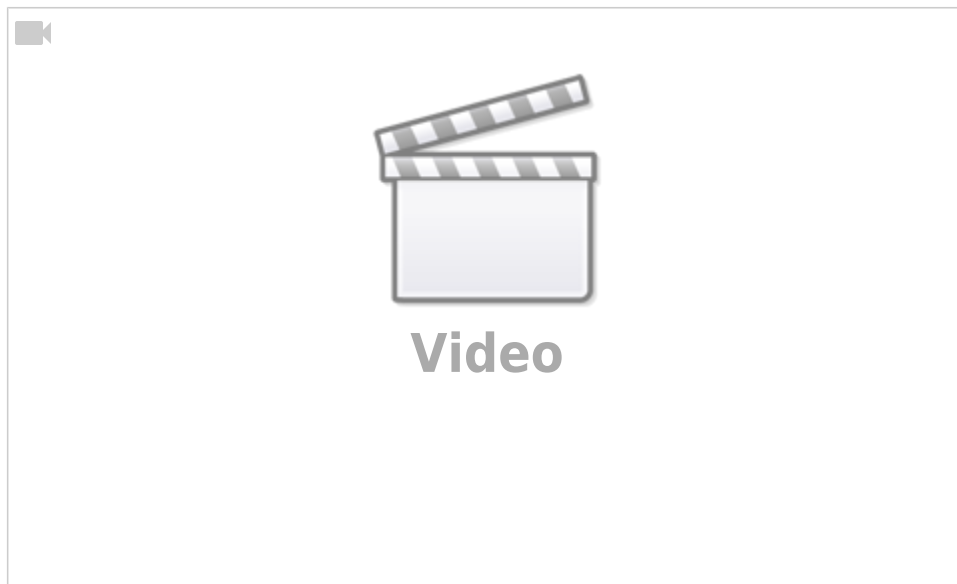


The steps are to:

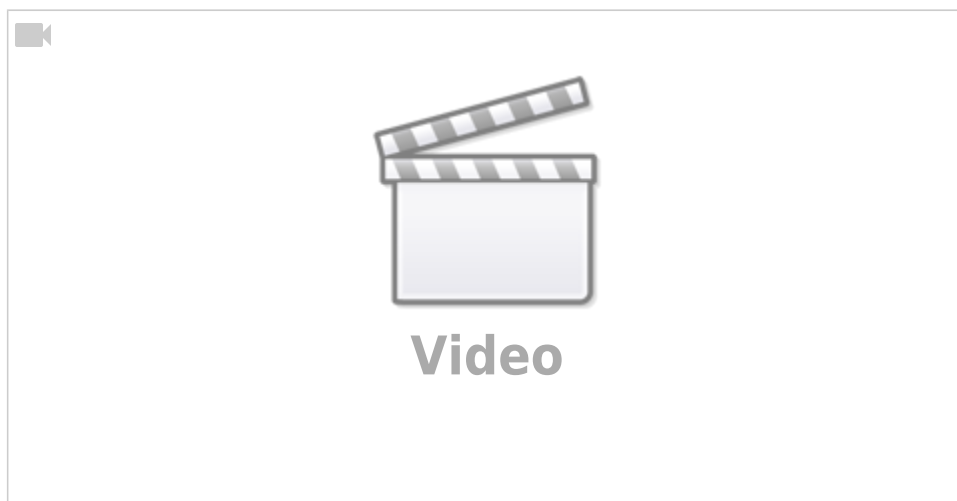
- Fill out the Preamble sheet.
- Log the contacts in the Log sheet.
- Export to Cabrillo using the button on the Preamble sheet.

Here are a few introductory videos I made to explain how to use the spreadsheet, which should also work with the RAC Winter Contest in December.

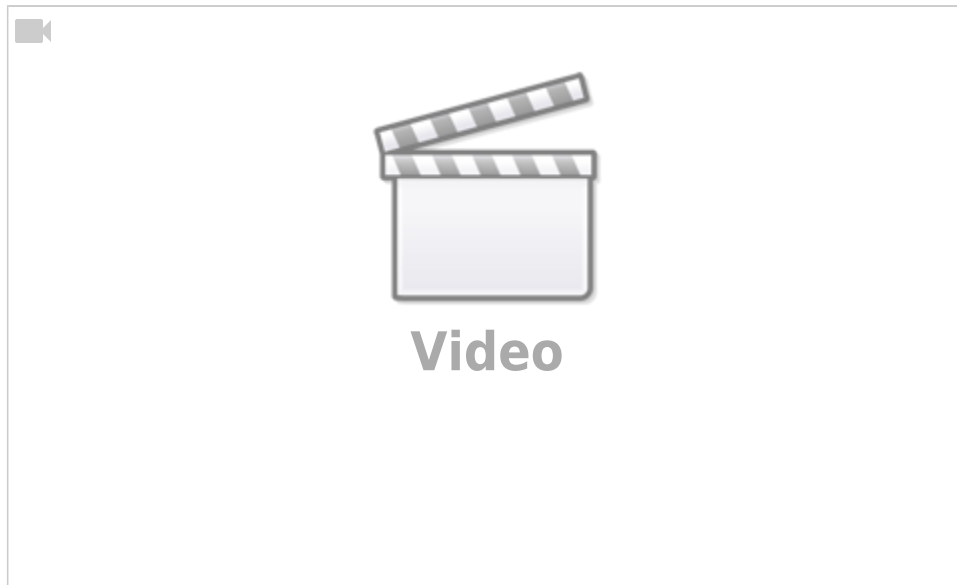
- The first video shows where to download [LibreOffice](#), and how to set the Security settings to allow LibreOffice to run macros: Tools → Options... → LibreOffice → Security → Macro Security → Medium



- The second video shows how to use the spreadsheet during the contest, and how to export the Cabrillo file after.



- The third video shows a bit of the hidden formulas and the script that generates the Cabrillo file. It's like looking under the hood of the car: it's not needed to drive it, and it doesn't really explain how to build a car either. But some might find it interesting.



Here's a copy of the code that generates the Cabrillo file:

```
REM ***** BASIC *****

sub cabrillo

'Get directory path from spreadsheet location. This will be used to create
./MyCallsign.log
Dim path as String
  GlobalScope.BasicLibraries.loadLibrary("Tools")
  path = Tools.Strings.DirectoryNameoutofPath(ThisComponent.url, "/" & "/"

'Define document and sheets
dim Doc as object
Doc = ThisComponent
Sheet1 = Doc.Sheets.getByName("Preamble")
Sheet2 = Doc.Sheets.getByName("Log")

'General Note: for getCellByPosition(x,y)      (0,0) = A1,  (1,0) = B1,  (0,1) = A2,
...

'Create Cabrillo file named "MyCallsign.log"
MyCallsign = UCase(Sheet1.getCellByPosition(1, 2).String)      'Operator's callsign
if MyCallsign = "" then                                         'File needs a
filename
  MyCallsign = "NoCallsign"
endif
filename = path & MyCallsign & ".log"

'Open MyCallsign.log and get ready to write to it
```

```

num = FreeFile()
open filename for output as #num

'Read "Preamble" sheet and create preamble of Cabrillo File
for i = 0 to 1          ' read first 2 rows as is.
    print #num, Sheet1.getCellByPosition(0, i).String & " " &
Sheet1.getCellByPosition(1, i).String
next

' Row 3 needs to be upper case.
print #num, Sheet1.getCellByPosition(0, 2).String & " " &
UCase(Sheet1.getCellByPosition(1, 2).String)

for i = 3 to 19          ' read the the rest up to row 20 as is.
    print #num, Sheet1.getCellByPosition(0, i).String & " " &
Sheet1.getCellByPosition(1, i).String
next

' Row 21 needs "@" before the callsign
print #num, Sheet1.getCellByPosition(0, 20).String & " @" &
UCase(Sheet1.getCellByPosition(1, 20).String)

Soapbox = Sheet1.getCellByPosition(1, 21).String
if Len(Soapbox) > 70 then
    truncated = "y"
    Soapbox = Left(Soapbox, 70)
    MsgBox("SOAPBOX message can have at most 70 characters. It was truncated to:" &
Chr(10) & Chr(10) & "" & Soapbox & "", 48)
endif

print #num, Sheet1.getCellByPosition(0, 21).String & " " & Soapbox

'Formatting Example. Comment out once finished
'    print #num,
"000000000011111111112222222222333333333344444444445555555555666666666677777777778"
'    print #num,
"1234567890123456789012345678901234567890123456789012345678901234567890"
'    print #num, "QS0:  1825 CW 2003-07-01 1044 VA1ABC          599 ON          VE4EAR
599 MB"
'    print #num, "QS0:  3510 CW 2003-07-01 1044 VA1AB          599 ON          K4BAI
599 103"
'    print #num, "QS0:  7155 PH 2003-07-01 1044 VE3KZ          599 ON          K5MM
599 005"
'    print #num, "QS0: 14205 PH 2003-07-01 1044 VE3KZ          599 ON          K4LTA
599 10"
'    print #num, "QS0: 21350 CW 2003-07-01 1044 VE3KZ          599 ON          K1EA
599 55"
'    print #num, "QS0: 28375 PH 2003-07-01 1050 VE3KZ          59 ON          VE5SF
59 SK"
'    print #num, "QS0: 50125 PH 2003-07-01 1055 VE3KZ          59 ON          VE3EJ
59 ON"
'    print #num, "QS0:146520 PH 2003-07-01 1055 VE3KZ          59 ON          VE3CZ

```

59 ON"

```

MyCallsign = MyCallsign & space(14 - Len(MyCallsign)) 'pad MyCallsign to make it
14 characters long.

MyProvince = Sheet1.getCellByPosition(1, 3).String 'my province
MyProvince = MyProvince & space(7 - Len(MyProvince)) 'pad MyProvince to 7
characters

ContestDate = Sheet1.getCellByPosition(1, 22).String 'contest date

'read "Log" Sheet starting at third row.
i = 2 'first row is i = 0 so
third row is i = 2
UTC = "anything" 'initial non-empty
condition for knowing when to stop loop
while UTC <> "" 'while the time entry is
not empty, process each row
    Freq = Sheet2.getCellByPosition(0, i).String 'frequency in column A (x =
0)
    Freq = Format(Freq, "0") 'round to the nearest
integer
    Freq = space(6 - Len(Freq)) & Freq 'pad frequency to 6 digits

    UTC = Sheet2.getCellByPosition(1, i).String 'time in column B (x = 1).
Also used to stop loop.

    Callsign = Sheet2.getCellByPosition(2, i).String 'callsign in column C (x =
2)
    Callsign = Callsign & space(13 - Len(Callsign)) 'pad Callsign to 13
characters

    RST = Sheet2.getCellByPosition(3, i).String 'received Signal Report in
column D (x = 3)
    RST = space(3 - Len(RST)) & RST 'pad RST to 3 digits

    Exch = Sheet2.getCellByPosition(4, i).String 'received Exchange in
column E (x = 4)

    RSTGiven = Sheet2.getCellByPosition(5, i).String 'received Signal Report in
column D (x = 3)
    RSTGiven = space(3 - Len(RSTGiven)) & RSTGiven 'pad RSTGiven to 3 digits

    Mode = Sheet2.getCellByPosition(6, i).String

'create line to print from "Log" sheet variables. The output should be
something like this:
'QS0: 1825 CW 2003-07-01 1044 VA1AB 599 ON VE4EAR 599 MB
'QS0: 14165 PH 2003-07-01 1044 VA1ABC 59 ON K7AB 59 MB
'QS0:146520 PH 2003-07-01 1044 VA1ABC 59 ON VE4EA 59 MB

```

```

logline = "QSO:"
logline = logline & Freq & " "
logline = logline & Mode & " "
logline = logline & ContestDate & " "
logline = logline & UTC & " "
logline = logline & MyCallsign
logline = logline & RSTGiven & " "
logline = logline & MyProvince & " "
logline = logline & Callsign & " "
logline = logline & RST & " "
logline = logline & Exch

print #num, Ucase(logline)                                'print logline to
text file

i = i + 1                                                  'Next row
UTC = Sheet2.getCellByPosition(1, i).String               'Look ahead to next time
entry to see if it's empty or not.

'      MsgBox(MyCallsign)
wend

print #num, "END-OF-LOG:"

close #num

msgbox ("Cabrillo log file has been created here:  " & Chr(10) & Chr(10) &
filename)

end sub

sub License

' This spreadsheet and its script by Patrick Truchon <va7fi@rbox.me> is licensed under
a
' Creative Commons Creative Commons Attribution-Share Alike 4.0 Unported License.
' <https://creativecommons.org/licenses/by-sa/4.0/>.
'
' You are free to:
' * Run them for any purpose.
' * Study and modify them.
' * Copy them to help others.
' * Improve them, and release the improvements to the public, so that
'   the whole community benefits.
'
' Provided that you:
' * Attribute the work to me by linking to
<https://scarcs.ca/howto/contestspreadsheets>
' * Distribute any derivative work under the same license.

```

end sub

Winter Field Day Contest

The [Logging Spreadsheet](#) was last updated on Jan 28, 2023 for the Winter Field Day on Jan 28-29, 2023.

Everything you need to know about the Winter Field Day contest is here: <https://www.winterfieldday.com>

Logs

During the contest, the following information must be logged for each station:

- Date and Time in UTC (for example 2023-01-28 and 1901)
- Frequency in kHz (for example, 14.125 MHz should be entered as 14125)
- Callsign
- Class (ex. 1H: 1 operator at home, 2O: 2 operators outside, 3I: 3 operators inside, 4M: 4 operators mobile)
- ARRL Location (ex. BC, AB, SF, ...)
- Mode (CW, DG, or PH)

At the end of the contest, the Cabrillo file must be submitted online at www.winterfieldday.com

7300 Recorded Voice TX

The other thing I learned during the contest is how to record a short message on my IC-7300 and play it back on the air so I could save my voice a bit. The details are on [Section 7 of the IC-7300 full manual](#):

I ended up recording three messages:

1. "CQ Canada Day Contest, this is VA7FI" (in phonetics)
2. "VA7FI" (in phonetics)
3. "Thank you, you are 59, BC"

To call CQ, I used the first message on a 7 second repeat loop. If someone answered, I could either press the button again to stop the loop, or use the PTT to jump in and acknowledge the station.

I used the second message to answer CQ calls during pile ups. All I'd have to do is press the button and hope I'd get an answer. This was great in the early morning when Justine was still in sleeping. With my headphones on, the whole thing was virtually silent.

I used the third message to give my exchange once I received the other station's exchange.

I still had to use the mic a bit depending on the situation, but these pre-recorded messages took care of a lot of the grunt work.

License

Both spreadsheets are licensed under Creative Commons [By-Sa](#) so you are free to:

- Run it for any purpose.
- Study and modify it.
- Copy it to help others.
- Improve it, and release the improvements to the public, so that the whole community benefits.

Provided that you:

- Attribute the work to me by linking to <https://scarcs.ca/howto/contestspreadsheets>
- Distribute any derivative work under the same license.