RAC Contests Log

This year was my first time participating in the RAC Canada Day contest on my own. It was a lot of fun and also very instructive. Here are a few things I learned.

First, everything you need to know about the RAC contests (either the Canada Day or the Winter Day) is located on the RAC website.

Logs

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 formatted as Cabrillo.

A Cabrillo file is really just plain text file formatted in a very specific way. For more information, see:

- This RAC pdf, or
- The WWROF website

For example, here's a copy of my Cabrillo file (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!
QSO: 14186 PH 2020-07-01 0008 VA7FI          59 BC      VE3PJ          59 ON
QSO: 14198 PH 2020-07-01 0012 VA7FI          59 BC      WB0TEV         59 14
QSO:146520 PH 2020-07-01 0112 VA7FI          59 BC      VE7DX          59 BC
QSO: 14211 PH 2020-07-01 0139 VA7FI          59 BC      N800           59 201
QSO: 14152 PH 2020-07-01 0148 VA7FI          59 BC      VE2CJR         59 QC
QSO: 14165 PH 2020-07-01 0159 VA7FI          59 BC      VE6RAC         59 AB
```
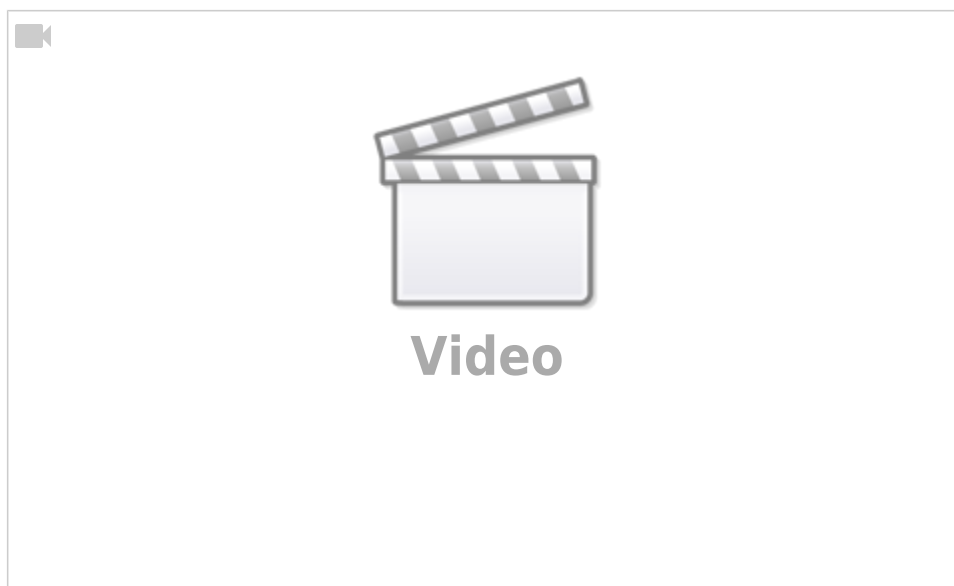
```
END-OF-LOG:
```

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.

RAC does NOT want this emailed to them in the body of the email. Instead, this should be saved in a plain text file named (in my case): VA7FI.LOG and attached to the email.
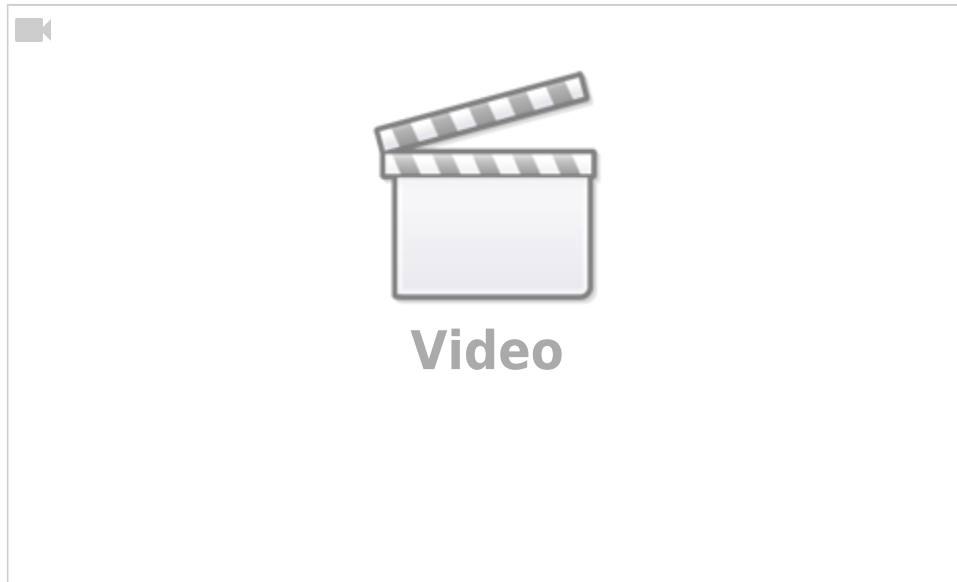
Regular contesters use logging programs which can generate these files while providing many bells and whistles during the contest to facilitate the logging process. Many people have recommended N1MM. RAC also has its own Microsoft Windows contest program.

But since I'm not a regular contester and I didn't have time to familiarize myself with a logging program that would run under GNU/Linux, I simply used a spreadsheet to log my entries and tweaked it as the day went on to add my points and check for duplicate stations. A few days after the contest was over, I cleaned up the spreadsheet and added some code to export the Cabrillo file the way RAC needs it.
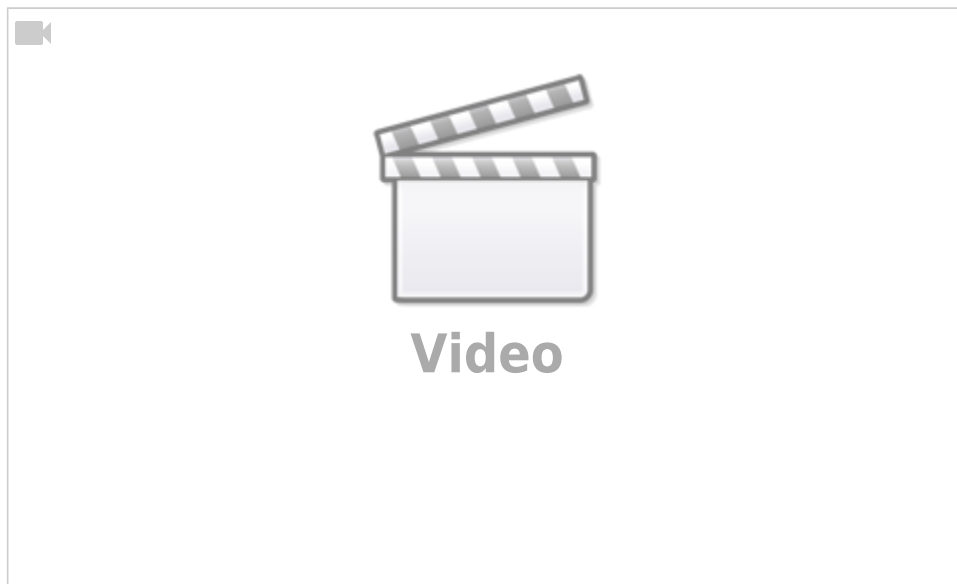
Here are a few introductory videos I made to explain how to use the spreadsheet, which should also work with the RAC Winter Contest in December.

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 it 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 is the link to](#)

download it

.

I've licensed it under a Creative Commons [By-Sa](#) so you are free to:

- Run the scripts for any purpose.
- Study and modify the scripts.
- Copy the scripts to help others.

- Improve the scripts, 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://ptruchon.pagekite.me>
- Distribute any derivative work under the same license.

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

```basic
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("Log")
    Sheet2 = Doc.Sheets.getByName("Preamble")


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

'Create Cabrillo file named "MyCallsign.log"
    MyCallsign = Sheet2.getCellByPosition(1, 2).String     'Operator's callsign
    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 18         ' read first 19 rows as is.
        print #num, Sheet2.getCellByPosition(0, i).String & " " &
Sheet2.getCellByPosition(1, i).String
    next
    ' Row 20 needs "@" before the callsign
    print #num, Sheet2.getCellByPosition(0, 19).String & " @" &
Sheet2.getCellByPosition(1, 19).String

    Soapbox = Sheet2.getCellByPosition(1, 20).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, Sheet2.getCellByPosition(0, 20).String & " " & Soapbox

'Formatting Example.  Comment out once finished
'    print #num,
"00000000011111111112222222222333333333344444444445555555555666666666677777777778"
'    print #num,
"123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890"
'    print #num, "QSO:  1825 CW 2003-07-01 1044 VA1ABC        599 ON      VE4EAR
599 MB"
'    print #num, "QSO:  3510 CW 2003-07-01 1044 VA1AB         599 ON      K4BAI
599 103"
'    print #num, "QSO:  7155 PH 2003-07-01 1044 VE3KZ         599 ON      K5MM
599 005"
'    print #num, "QSO: 14205 PH 2003-07-01 1044 VE3KZ         599 ON      K4LTA
599 10"
'    print #num, "QSO: 21350 CW 2003-07-01 1044 VE3KZ         599 ON      K1EA
599 55"
'    print #num, "QSO: 28375 PH 2003-07-01 1050 VE3KZ          59 ON      VE5SF
59 SK"
'    print #num, "QSO: 50125 PH 2003-07-01 1055 VE3KZ          59 ON      VE3EJ
59 ON"
'    print #num, "QSO: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 = Sheet2.getCellByPosition(1, 3).String      'my province
   MyProvince = MyProvince & space(7 - Len(MyProvince))    'pad MyProvince to 7
characters

   ContestDate = Sheet2.getCellByPosition(1, 21).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 emtry is
not empty, process each row
      Freq = Sheet1.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 = Sheet1.getCellByPosition(1, i).String          'time in column B (x = 1).
Also used to stop loop.

      Callsign = Sheet1.getCellByPosition(2, i).String     'callsign in column C (x =
```

```
2)
      Callsign = Callsign & space(13 - Len(Callsign))       'pad Callsign to 13
characters

      RST = Sheet1.getCellByPosition(3, i).String           'received Signal Report in
column D (x = 3)
      RST = space(3 - Len(RST)) & RST                       'pad RST to 3 digits

      Exch = Sheet1.getCellByPosition(4, i).String          'received Exchange    in
column E (x = 4)

      RSTGiven = Sheet1.getCellByPosition(5, i).String      'received Signal Report in
column D (x = 3)
      RSTGiven = space(3 - Len(RSTGiven)) & RSTGiven        'pad RSTGiven to 3 digits

      Mode = Sheet1.getCellByPosition(6, i).String

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

      i = i + 1                                             'Next row
      UTC = Sheet1.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 <https://ptruchon.pagekite.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://ptruchon.pagekite.me>
'    * Distribute any derivative work under the same license.

end sub
```