

```

REBOL[MY TEST]
System/ports/serial: [ com5 ] ;if using linux this line is slightly different
ser: open/direct/no-wait serial://port1/250000/none/8/1
ser/rts-cts: false
update ser
buffer: make string! 1
logFile: "Hx19log.txt"
bb: 0 bc: 0 tt: ""

;Following function computes xx string checksum and appends /hexsum to the string
;where hexsum is the hexadecimal representation of the sum of the string characters
hx19: funct [xx]
[
  pp: copy xx
  ss: 0
  foreach char pp [ss: ss + char]
  hx: to-hex ss
  until [sn: take hx sn > #"0"]
  insert hx sn
  append pp "/" append pp hx append pp CR
  return pp
]
;following function sets the display layout with check buttons and etc.
view/new/title layout
[
  across
    label "SYNC"
    check [
      either bb > 0
        [bb: 0 insert ser hx19 "M&%" update ser]
        [bb: 1 insert ser hx19 "M&$" update ser]
      ]
    label "LOG"
    check [
      either bc > 0
        [bc: 0 close log]
        [bc: 1 log: open/new %hx19log.txt]
      ]
  f: field 200
  btn "TX" [insert ser hx19 f/text update ser]
  btn "SetUP File"
  [
    px: read/lines %hx19setup.txt
    repeat nn 20
    [
      kk: pick px nn
      if 0 = length? kk [break]
      print kk
      insert ser hx19 kk update ser
      wait 0.05
    ]
  ]
  ] "HX19 ACCESS"
insert-event-func [switch event/type [close [close ser if bc > 0 [close log] quit]]event]
;the following code monitors the incoming data from the serial port and prints in a popup
box
forever [
until [
  while[empty? buffer][read-io ser buffer 1 wait 0.0001]
  x: to-integer first buffer
  append tt buffer
  clear buffer
  x = 13
]
print tt
if bc > 0 [append log tt]
clear tt
]

```