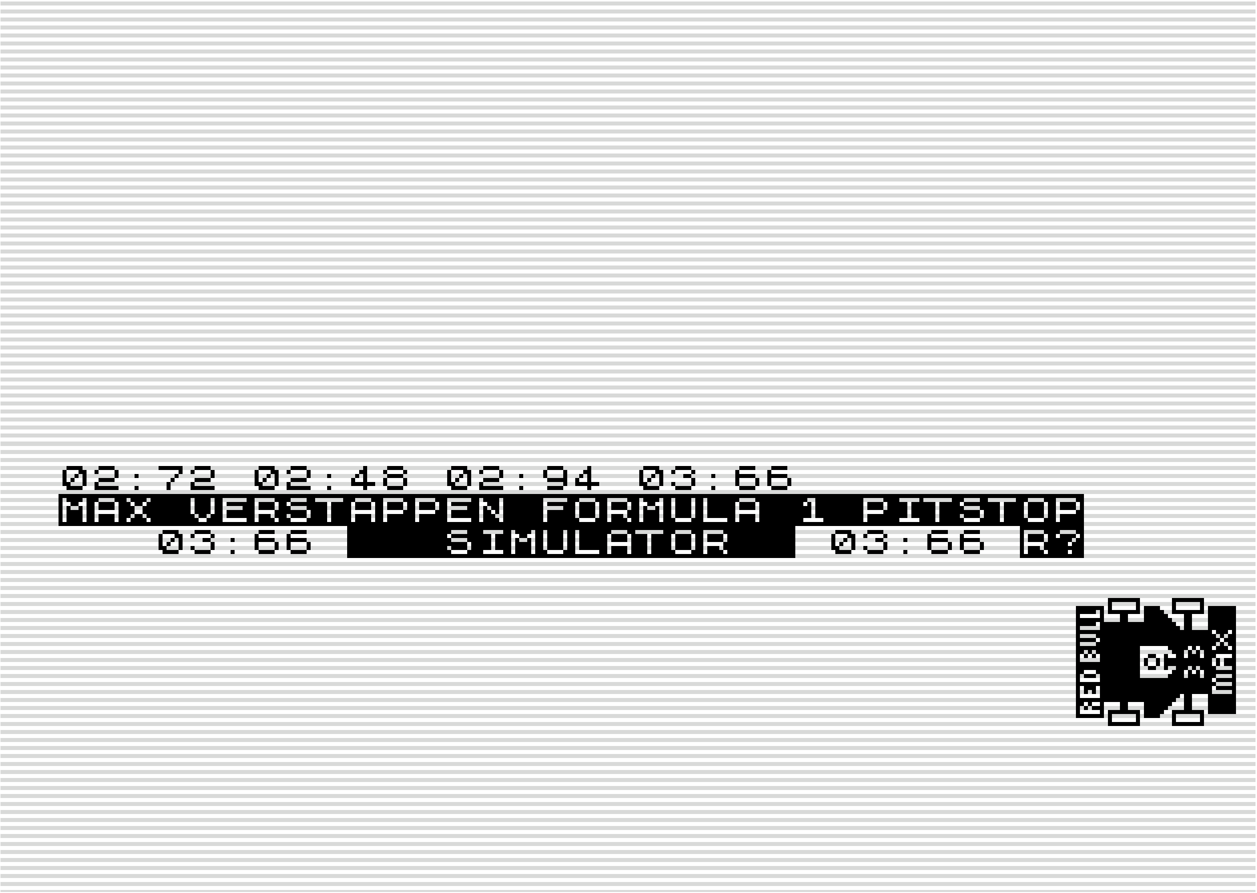
Max Verstappen Formula One Pitstop Simulator



```
02:72 02:48 02:94 03:66
MAX VERSTAPPEN FORMULA 1 PITSTOP
   03:66    SIMULATOR    03:66 R?
```

I got this idea when I saw a broken trucktyre besides the road. I once made another game that recorded input. This game doesn't record input but the time each state a wheel is in. This method made it possible to fit it in 1K.

```
; Max Verstappen Formula One Pitstop Simulator
; Game 70 in 1K hires for the ZX81.

? * TORNADO *

           ORG  #4009             ;#4009
           DUMP 49161

; program starts here, both BASIC and machinecode
basic      EX   AF,AF'            ; delay intrupt,opcode no bit6
           LD   H,B               ; preset for 48K bug to #40
           JR   init0             ; continue where room

           DEFB 236,212,28        ; The BASIC
           DEFB 126               ; fully placed over sysvar
           DEFB 143,0,18          ; start BASIC=#4009 also MC

eline      DEFW last              ; needed by loading
chadd      DEFW last-1
xptr       DEFW 0
stkbot     DEFW last
stkend     DEFW last
berg       DEFB 0
mem        DEFW 0
           DEFB 128
           DEFB 0,0,0

; all above reusable AFTER loading

lastk      DEFB 255,255,255       ; used by ZX81
margin     DEFB 55                ; used by ZX81
nxtlin     DEFW basic             ; reusable after load

init0      LD   IX,hr             ; hr lowbyte bit 5 reset
; lowbyte over flagx which resets bit 5 on load
; HR must be set on right address or game crashes

           LD   E,L               ; DE now #xx.L

taddr      DEFW 0                 ; used by ZX81 on LOAD only
                                  ; unharmed code

           LD   B,4               ; copy >1K code

frames     DEFB #16+1             ; LD D,n , after LOAD -1
           DEFB #C0               ; highbyte must have bit 7 set
coords     LDIR                   ; DE now #C0.L  = Hl +#8000
                                  ; fix 48K bug before display
prcc       JP   init              ; continue to mainprog

cdflag     DEFB 64                ; used by ZX81

; Place ANY code to fill up to #4040
hrdelay    PUSH AF                ; filler
           POP  AF                ; when display is not
           NOP                    ; altered
           NOP
           EX   (SP),HL
           EX   (SP),HL
           JR   showline


; some lowres, HR must start AFTER #403F
hr         LD   HL,lowres+#8000   ; the lowres display
           LD   BC,#469           ; minimum needed #11
           LD   A,#1E
           LD   I,A
           LD   A,#FB
           CALL #2B5              ; show lowres screen


           LD   HL,disptab
           LD   A,car/256
           LD   I,A

           CALL setlbuf           ; display team1 and team2

           PUSH HL
           POP  HL
           PUSH HL

st1        LD   HL,gr0            ; status team1
st2        LD   DE,gr0            ; status team2
           CALL showteam          ; show status team1 and team2
           POP  HL

           PUSH HL                ; outline car with teams
           POP  HL
           EX   (SP),HL
           EX   (SP),HL

backcar    CALL #43FF             ; drive delay routine

           LD   C,car*256/256

           LD   B,32              ; car is 32 lines

loop       LD   A,(HL)            ; test alter wheel display
           CP   B
           JR   NZ,hrdelay        ; if not, sync routine

           INC  L
           CALL setlbuf           ; show carstatus

showline   LD   A,C               ; get cardata
           CALL lbuf+#8000        ; show cardata
           ADD  A,5               ; point to next cardata
           LD   C,A

           PUSH HL                ; timer
           POP  HL

           DJNZ loop              ; show full car

           CALL setlbuf           ; display team3 and team4

           DEC  SP                ; filler
           DEC  SP
           POP  HL

frontcar   CALL #43FF-32          ; drive delay routine

st3        LD   HL,gr0            ; status team3
st4        LD   DE,gr0            ; status team4
           CALL showteam          ; show status team3 and team4

; fixed end of HR-routine
exit       CALL #292              ; back from intrupt
           CALL #220
           LD   IX,hr
           JP   #2A4

showteam   LD   BC,teamline+1

teamstart  DEFB #DD               ; ld ixh,8
           LD   H,8

l1del      LD   A,(HL)            ; get UDG team1/team3
           LD   (BC),A            ; write status
           INC  BC
           INC  BC
           LD   A,(DE)            ; get udg team2/team4
           LD   (BC),A            ; write status
           DEC  BC
           DEC  BC                ; back to pointer team 1/3
           INC  L                 ; next udg data 1/3
           INC  E                 ; next udg data 2/4
           LD   A,teamline*256/256
           CALL lbuf+#8000        ; show data

           PUSH HL                ; filler
           POP  HL
           EX   (SP),HL
           EX   (SP),HL

           DEFB #DD               ; dec ixh
           DEC  H
           RET  Z                 ; filler and endstate
           JR   l1del             ; continue loop

gr2        DEFB 24,40,79,129,129,79,40 ; arrow left
gr3        DEFB 24,20,242,129,129,242,20 ; arrow right
gr1        DEFB 24,36,66,129,231,36,36 ; arrow up
gr4        DEFB 60,36,36,231,129,66 ; arrow down
disptab    DEFB 36,24
t12        DEFB 32,0,0            ; wheel1 or wheel2
           DEFB 28,0,0            ; show car
t34        DEFB 4,0,0             ; wheel3 or wheel4
gr0        DEFB 0,0               ; display team3 or team4
           DEFW 0,0,0

setlbuf    LD   DE,p1             ; destination1
           LD   A,(HL)            ; get show on or show off
           LD   (DE),A            ; set on/off
           INC  L
           LD   E,p2*256/256      ; destination2
           LD   A,(HL)            ; same second wheel
           LD   (DE),A
           INC  L
           RET

delay      LD   HL,frames         ; standard delay routine
           ADD  A,(HL)
wfr        CP   (HL)
           JR   NZ,wfr
           RET

sp41       EQU  #4108-$           ; 1 left
           DEFS sp41

; for correct display car must start on #4108
car        DEFB 0,255,0,255,0
           DEFB 0,129,0,129,0
           DEFB 254,129,120,129,127
           DEFB 250,255,124,255,127
           DEFB 130,24,126,24,127
           DEFB 254,24,127,24,127
           DEFB 251,255,255,152,127
           DEFB 131,255,255,216,127
           DEFB 255,255,255,255,221
           DEFB 135,255,255,255,235
           DEFB 251,255,255,255,247
           DEFB 135,255,255,255,235
           DEFB 255,255,1,245,221
           DEFB 215,255,0,234,255
           DEFB 171,255,50,110,225
           DEFB 131,255,75,255,215
           DEFB 255,255,75,255,215
           DEFB 255,255,50,117,225
           DEFB 199,255,0,234,255
           DEFB 187,255,1,238,225
           DEFB 131,255,255,255,223
           DEFB 255,255,255,255,225
           DEFB 187,255,255,255,223
           DEFB 171,255,255,255,193
           DEFB 131,255,255,216,127
           DEFB 255,255,255,152,127
           DEFB 210,24,127,24,127
           DEFB 174,24,126,24,127
           DEFB 130,255,124,255,127
           DEFB 254,129,120,129,0
           DEFB 0,129,0,129,0
           DEFB 0,255,0,255

teamline   DEFB 0,0,0,0,0

lbuf       LD   R,A
           DEFB 0
p1         DEFB 0,0
p2         DEFB 0,0
           RET

eog        LD   HL,time-1         ; your time
           LD   DE,fasttime-1     ; fastest time
           LD   BC,6              ; size 5
fihi       INC  HL
           INC  DE
           DEC  C                 ; when C=0
           LD   A,(DE)            ; (de) = #76
           CP   (HL)              ; (hl) = #7f
           JR   Z,fihi            ; so not same
           CALL NC,#19F9          ; and no fast time
           LD   HL,shwrep
           LD   (HL),B            ; show REPLAY ?
           PUSH HL                ; save for repair
wkey       LD   BC,(lastk)
           LD   A,C
           INC  A
           JR   Z,wkey
           CALL #7BD              ; translate pressed key
           POP  HL
           LD   (HL),118          ; set NewLine back
           SUB  13
           LD   BC,#405           ; 4 wheels record, player away
           JR   Z,replay

start      LD   A,(lastk)         ; game over, wait for
           SUB  %10111111         ; newline
           JR   NZ,start

           LD   BC,#101           ; 1 wheel which is player

replay     DEC  A                 ; set car before pitlane
           LD   (backcar+1),A     ; take off delay behind car
           SUB  32
           LD   (frontcar+1),A    ; add delay in front of car

           LD   HL,time4+4        ; teamtimes on screen
cls        LD   (HL),0            ; will be erased
           DEC  HL
           LD   A,(HL)
           CP   118
           JR   NZ,cls

           PUSH BC
           CALL movecar           ; drive into pitlane

           POP  BC                ; first 1 team
playloop   LD   HL,timecount
           LD   DE,#404
```

```
setcnt      LD   (HL),1            ; initial counter needed
            INC  HL
            DEC  D
            JR   NZ,setcnt

            PUSH BC

            LD   BC,timetab+3
            LD   HL,cartab+3
setcartab   DEC  E
            LD   A,E
            ADD  A,A               ; e*2
            ADD  A,A               ; e*4
            ADD  A,A               ; e*8
            LD   (HL),A
            EXX
            LD   E,A
            LD   D,#40
            LD   A,(DE)
            EXX
            LD   (BC),A

            DEC  HL
            DEC  BC
            JR   NZ,setcartab      ; reset all pointers
            POP  BC
            INC  DE                ; D=0, E=1

            LD   HL,#1C1C          ; "00"
            LD   (time),HL         ; seconds set to "00"
            LD   (time+3),HL       ; hundredssec set to "00"

playteam    PUSH BC                ; team not yet ready

            XOR  A
            LD   (cntready+1),A    ; reset teams ready counter

settime     DEC  A
            CALL delay             ; wait 1/50 sec

            LD   HL,time+4         ; update time
            INC  (HL)              ; 1/100 sec
settime2    INC  (HL)              ; 1/50 sec
            LD   A,(HL)
            CP   38
            JR   NZ,allteams
            LD   (HL),28
findtime    DEC  HL
            LD   A,(HL)
            CP   28
            JR   C,findtime
            JR   settime2

allteams    LD   HL,cartab-1       ; each wheel, what status?
            LD   A,B
            ADD  A,L
            LD   L,A
            LD   (curpos+1),HL

            LD   A,(HL)            ; get status current wheel
            AND  7                 ; make status all the same
            CP   6                 ; 6 is wheel replaced
            LD   A,B               ; preload for not stop and
            JR   Z,stop1           ; stop, count wheel ready
```

```
           CP   C                 ; not stop, check player
           JR   NZ,checkrec       ; no match is recording

           LD   A,(HL)            ; get current pointer
           AND  7
           CP   5
           JR   NZ,keepe
           LD   E,1               ; signal ready 1 repeat
keepe      LD   A,(HL)

           ADD  A,keyteamtab*256/256
           LD   L,A               ; which key must be pressed
           LD   A,(HL)            ; get key to press

; A holds key to press
; read keyboard, no match inc rec
; match, timerloop
           EXX
           LD   BC,(lastk)
           LD   A,C
           INC  A
           LD   A,C
           CALL NZ,#7BD
           EXX
           INC  D                 ; add 2/100 recording time
           CP   (HL)
           JR   Z,matchkey
set20      LD   E,21              ; penalty delay for errorkey

matchkey   DEC  E
           JR   NZ,dispnxteam

curpos     LD   HL,cartab
           INC  (HL)              ; next step team
           LD   L,(HL)            ; get next timerec position
           LD   A,L
           DEC  L                 ; undo inc
           LD   H,#40             ; set highbyte recordtable
           LD   (HL),D            ; write time done step
           CALL showkey
; show right key to press
           LD   D,0               ; recorder to zero
           JR   set20             ; 20 steps for next key

STOP1      LD   HL,cntready+1
           INC  (HL)              ; another wheel changed
           ADD  A,A
           ADD  A,B
           ADD  A,A               ; which team?, B*6
           EXX
           LD   HL,time
           LD   DE,time1-6        ; time on sreen per team
           ADD  A,E
           LD   E,A               ; now correct team
           LD   A,(DE)            ; check already shown
           OR   A
           LD   BC,5
           CALL Z,#19F9           ; set only when later time

           JR   dispnxteam-1      ; check steps done

checkrec   LD   A,B
           EXX
           LD   HL,timetab-1
           ADD  A,L
           LD   L,A
           LD   A,(HL)            ; get played time
           DEC  (HL)
           JR   NZ,dispnxteam-1   ; end of recorded, next step

; set next time
           EX   DE,HL
           LD   HL,(curpos+1)     ; replay recording
           INC  (HL)
           LD   L,(HL)
           LD   H,#40
           LD   A,(HL)
           LD   (DE),A
           CALL showkey-2         ; 2 opcodes there to keep a JR
           EXX
           EXX

dispnxteam DJNZ allteams          ; DJNZ, no room left to code

           POP  BC

cntready   LD   A,0
           CP   B

           JP   NZ,playteam       ; check current teams ended

           INC  B                 ; add a team/wheel
           LD   A,B
           CP   5                 ; all wheels played?
           LD   C,B               ; next wheel for player
           JP   C,playloop        ; play 4 teams

           LD   HL,eog
           PUSH HL                ; end of game as return

movecar    LD   B,16              ; first drive in then out
drive      LD   HL,backcar+1
           DEC  (HL)              ; add time behind car
           LD   L,frontcar*256/256+1
           INC  (HL)              ; take off same time in front
           LD   A,252
           CALL delay
           DJNZ drive
           RET                    ; into game or game over

           LD   A,L
           EXX
showkey    LD   HL,keyteamtab
           ADD  A,L
           LD   L,A
           LD   A,(HL)
           DEC  HL
           LD   L,(HL)
           PUSH HL
           PUSH DE
           LD   L,keyteamtab*256/256
           LD   DE,key2char-1
fchar      INC  HL
           INC  DE
           CP   (HL)
           JR   NZ,fchar
           LD   HL,team2pos-1
           LD   A,B
           ADD  A,L
           LD   L,A
           LD   L,(HL)
           LD   H,st1/256
           LD   A,(DE)
           LD   (HL),A
           POP  DE
           POP  HL
           LD   A,L
           CP   11
           RET  NC
           LD   HL,t12
           LD   A,B
           CP   3
           JR   C,hlfnd
           LD   L,t34*256/256-2
hlfnd      ADD  A,L
           LD   L,A
           LD   A,(HL)
           XOR  H
           LD   (HL),A
           RET

x          EQU  101

lowres     DEFB 118
time1      DEFW 0,0,0
time2      DEFW 0,0,0
time3      DEFW 0,0,0
time4      DEFB 0,0,0,0,0
           DEFB 118

           DEFB "M"+x,"A"+x,"X"+x,128,"V"+x,"E"+x,"R"+x,"S"+x
           DEFB "T"+x,"A"+x,"P"+x,"P"+x,"E"+x,"N"+x,128

           DEFB "F"+x,"O"+x,"R"+x,"M"+x,"U"+x,"L"+x,"A"+x,128
           DEFB 157,128           ; inverted 1

           DEFB "P"+x,"I"+x,"T"+x,"S"+x,"T"+x,"O"+x,"P"+x

           DEFB 118,0,0,0
time       DEFB 35,28,14,35,28,#7F; same time = slower marker
           DEFB 128,128,128
           DEFB "S"+x,"I"+x,"M"+x,"U"+x,"L"+x,"A"+x,"T"+x
           DEFB "O"+x,"R"+x,128,128,0
fasttime   DEFB 35,28,14,35,28
shwrep     DEFB 118
           DEFB "R"+x,143,118

key2char   DEFB gr1*256/256       ; up
           DEFB gr2*256/256       ; left
           DEFB gr3*256/256       ; right
           DEFB gr4*256/256       ; down
           DEFB gr0*256/256       ; space

keyteamtab DEFB 255,10,26,25,5,255,255,0 ; team1 .ulrd.
           DEFB 255,10,25,26,5,255,255,0 ; team2 .urld.
           DEFB 255,5,26,25,10,255,255,0 ; team3 .dlru.
           DEFB 255,5,25,26,10 ,255 ,255 ; team4 .drlu.
```

```
team2pos   DEFB st1*256/256+1
           DEFB st2*256/256+1
           DEFB st3*256/256+1
           DEFB st4*256/256+1

init       LD   A,201             ; opcode RET
           LD   HL,#43FF          ; built delay line
           LD   B,33              ; at emd of memory
setline    LD   (HL),A
           XOR  A
           LD   SP,HL             ; set stack back
           DEC  HL
           DJNZ setline
           JP   start             ; start the game

vars       DEFB 128
?
last       EQU  $

timecount  EQU  vars              ; each team remaining time
cartab     EQU  timecount+4
timetab    EQU  #4021
```