

The GenerAlly TrackFile 1.05

3/2010 majortom

The Trackfile is devided into two major parts

T1 Changes by Trackeditor LandMap,HMap,Objects

T2 Changes by Racing Besttime,Cars,Ghost

T1:

T11: Header+Settings

T12: LandMap

T13: HMap

T14: Objectlist

T15: Pits

T16: Checkpoints

T17: AI-Line

T18: Pit-Line

T19: Author

T1A: Comment

T1B: Tracklength

T2:

T21: Besttimes

T22: Drivernames

T23: Cars

T24: Date/Time of Besttimes

T25: Ghost

(The following values are an example out of my track Ascheloh)

T1: LandMap,HMap,Objects

T11: Header+Settings

Address: Value Hex

0000: 05 2 Byte Version 05 01 = 1.05

0001: 01

0002: 47 2 Byte ASC(47)=G ASC(52)=R =GR

0003: 52

0004: 53 4 Byte lenghts of T1 = start address of T2

0005: 96 = 00029653H = 169555 dec offset

0006: 02

0007: 00

0008: 8C worldsize 8CH = 140

0009: 00 waterlevel = 0

000A: 37 Angel 37H = 55 degree

000B: 00 Angel not used, maybe FF=negativ=Track up/down

000C: 00 Rotation 0 degree (example: F7 FF = 247-256 = -9)

000D: 00 Rotation 0 = positiv, FF = negativ

000E: 64 Zoom 64H = 100

000F: 00 Start/Finishline 0/1

T12: LandMap

The landmap is a list of compressed pixel. It starts left/down and runs line by line to up/right. Dimension is 512x512 px

One set of pixel consists of two bytes = 16 bit:

Bit 1-12 = number of pixel with same color

Bit 13-16 = index of color

example: F2 FF = FFF2 = 1111111111110010 = 4096 x Index 2
= 4096 pixel of green (4096 is maximum of compression)

Address: Value Hex

0010: 90 4 byte length of LandMap 00013B90H

0011: 3B = 80784 datasets, 1 set=2 bytes

0012: 01 Landmap = 161568 bytes

0013: 00

0014: 72 0072 = 0000000001110010 = 7 x Index 2

0015: 00 = 7 pixel of green

0016: 1C 001C = 0000000000011100 = 1 x Index 12

0017: 00 = 1 pixel of darkgrey/black

. .
. .

Index:

0=RGB(160,160,160)	;darkgrey
1=RGB(180,180,180)	;lightgrey
2=RGB(144,220, 96)	;green
3=RGB(190,200,128)	;greenbrown
4=RGB(150,140,100)	;brown
5=RGB(190,180,160)	;lightbrown
6=RGB(240,220,210)	;lightred
7=RGB(246,251,254)	;darkwhite
8=RGB(220,231,240)	;bluewhite
9=RGB(255,255,255)	;white
10=RGB(255,255, 0)	;yellow
11=RGB(255, 0, 0)	;red
12=RGB(63, 63, 63)	;darkgrey/black
13=RGB(0, 0, 0)	
14=RGB(0, 0, 0)	
15=RGB(0, 0, 0)	

The compression works best with large areas of same color.
This is the reason, because a track with simple landmap
is much smaller than a track with more details.

T13: HMap

The HMap is a list of $64 \times 64 = 4096$ values: 0=low 255=high
A square of 8×8 Pixel in LandMap belong to 1 value in HMap.
All squares with value $< \text{waterlevel}$ are filled with water.
The list runs again left/down to up/right

Address: Value Hex

```
27734: 00    2 byte lenght of HMap 00 10 = 1000H = 4096
27735: 10    always the same
27736: A4    high level = A4H = 164
27737: A4
      .    .
      .    .
```

Startaddress of HMap is $10H + 4 + 2 \times 13B90H = 27734H$
(Header + 4Byte lenght + lenght LandMap)

T14: Objects

Objectlist starts with 2 bytes of number of objects.
For each object follows a set of 8 bytes.

t1 t2 x1 xh y1 yh z1 zh Typ,x,y,z=rotation
x,y describes centre of object, dimension depending of worldsize
example: Bridge ws100 = 159×62 px , wall ws100 = 31×4 px

```
t1      Typ      t2      ?
xhx1    Position x 0-1024 corresponding 0-512 in LandMap
yhy1    Position y 0-1024 corresponding 0-512 in LandMap
zh       Rotation Angel of Object 0-F4=0-244=0-360 Degree
          Rotation clockwise starting at 3 o'clock
z1       Rotation 0-255 bow-minutes (not so important)
```

Typ t1:

00 Start	12 House
01 Concrete wall	13 Office Block
02 Soft wall	14 Stand
03 Hay	15 Boat
04 Palm tree	16 Pier
05 Leaf tree	17 Bridge
06 Bush	18 Flat Bridge
07 Traffic sign	19 Fir tree
08 Light	20 Pine tree
09 Stone	21 Cactus
10 SunkTyre	22 Gate
11 Both	23 Fence
	24 Concrete Post

I don't know the meaning of t2. It seems to be a random value changing after each new saving in Trackeditor. This is a reason because a savegame does not work with a track that is only new saved whitout changing. A different t2 value results a new and different checksum for the track.

Startaddress of Objectlist is HMap + 1002H = 27734H+1002H=28736H
(2 Bytes lenght)

Address: Value Hex

28736:	44	2 Byte number of objects
28737:	01	0144H=324 Objects
28738:	00	First Object 00=Start
28739:	36	??
2873A:	D8	x-pos 01D8H=472 $x = 472/2 = 236$
2873B:	01	
2873C:	C2	y-pos 03C2H=962 $y = 962/2 = 481$
2873D:	03	
2873E:	EF	bow-minutes %
2873F:	DB	Angel dBH=219 $219/244*360=323$ degree
.	.	(starting at 3 o'clock)
.	.	

T15: Pits

Pitlist starts with 1 byte of number of pits.

For each pit follows a set of 6 bytes: x1 xh y1 yh z1 zh

Rule for x,y,z (Rotation) is the same as objects (T14)

Startaddress of Pitlist is Objectlist+2+#Objects*8
(2Bytes lenght) 28736H+2+144H*8=29158H

Address: Value Hex

29158:	06	1 Byte number of Pits = 6
29159:	8E	x-pos 038EH=910 $x = 910/2 = 455$
2915A:	03	

T16: Checkpoints

Checkpointlist starts with 1 byte of number of checkpoints.

For each checkpoint follows a set of 8 bytes.

x1 xh y1 yh x1l x1h y1l y1h

Endpoints x,y - x1,y1 of Checkpoint (simple Line) 0-1024

Startaddress of Checkpoints is Pitlist+1+#Pits*6
(1Byte lenght) 29158H+1+6*6 = 2917DH

Address: Value Hex

2917D: 08 1 Byte number of checkpoints = 8
2917E: E4 $02E4H=740$ $x= 740/2 = 370$
2917F: 02
29180: 60 $0260H=608$ $y= 608/2 = 304$
29181: 02
29182: 5E $025EH=606$ $x1= 606/2 = 303$
29183: 02
29184: 60 $0260H=608$ $y1= 608/2 = 304$
29187: DB

. .
. .

Checkpoint 1 = x,y - x1,y1 = 370,304 - 303,304

T17: AI-Line

AI-Line starts with 2 bytes of number of knots
For each knot follows a set of 4 bytes.

x1 xh y1 yh

Startaddress of AI-Line is Checkpoints+1+#CP*8
(1Byte lenght) $2917DH+1+8*8 = 291BEH$

Address: Value Hex

291BE: 87 2 Byte number of knots = $0087H = 135$
2917E: 00
2917F: 35 $0235H=565$ $x= 565/2 = 282$
29180: 02
29181: C3 $03C3H=963$ $y= 963/2 = 481$
29182: 03

. . AI-Line-Start x,y = 282,481
. .

A straight line is devided in several parts, so AI gets better direction to nearest knot.

T18: Pit-Line

Pit-Line starts with 1 byte of number of knots
For each knot follows a set of 4 bytes.

x1 xh y1 yh

Startaddress of Pit-Line is AI-Line+2+#Knots*4
(2Byte lenght) $291BEH+2+135*4 = 293DCH$

Address: Value Hex

293DC: 17 1 byte number of Knots = 17H = 23
293DD: FC 01FCH=508 x=508/2=254
2917F: 01
29180: DA 03DAH=986 y=986/2=491
29181: 03
 . Pit-Line-Start x,y=254,491
 .

1 byte lenght for Pit-Line means, it is limited to 255 knots.
This datablock ends with 2 bytes FF FF

T19: Author

A textblock with 30 bytes is reserved for the Author.
Bytes that are not used, are filled with 00.

Startaddress of author is Pit-Line+1+#Knots*4+2
(1Byte lenght, 2Byte FF FF) 293DCH+1+17H*4+2=2943BH

T1A: Comment

A textblock with 500 bytes is reserved for the Comment.
Bytes that are not used, are filled with 00.

Startaddress of comment is author+30 (1EH)
2943BH+1EH = 29459H

This datablock ends with 2 bytes FF FF

T1B: Tracklenght

4 Bytes at end of datablock T1 are for tracklenght

Startaddress of Tracklenght is comment + 500 (1F4H) +2 (FF FF)
29459H+1F4H+2 = 2964FH

Address: Value Hex

2964F: A7 000002A7H = 679m
2964F: 02
2964F: 00
2964F: 00

Maybe the second 2 bytes are for something other, I don't know.
They are always 00 00 because I never saw a track >65535m

Tracklenght + 4 is startaddress of datablock T2 = 29653H
This is the address found at byte 4 at beginnig of block T1.

T2: Besttime,Cars,Ghost

T21: Besttimes

T2 starts with 14 bytes of besttimes. This are 2 bytes for total besttime and 2 byte for each record 1-6. 70H 17H = 60.00

Address: Value Hex

29653: 45 Total besttime 0745H = 18.61

29654: 07

29655: 45 Record 1: 0745H = 18.61

29656: 07

. .
. .

T22: Drivernames

At address besttimes + 14 = 29653H + 14 (EH) = 29661H follows a list of drivernames. For each driver are 13 bytes reserved. Bytes not used are 00. Drivernames are corresponding to besttimes.

T22: Cars

At address drivernames + 13*7 = 29661H + DH*7 = 296BCH follows a list of cars corresponding to besttimes/drivernames. For each car are 1344 (540H) bytes reserved. This is the complete car file used in GeneRally without the first 8 bytes (Header of car file). The stored car file starts directly with the name of the car. Bytes not used are filled with 00.

T24: Date/Time of Besttimes

At address cars + 1344*7 = 296BCH + 540H*7 = 2BB7CH follows a list of Date/Time of Besttimes. 16 bytes are reserved for each date/time corresponding to total besttime and records 1-6. Bytes not used are filled with 00.

Address: Value Hex

2BB7C: D9 Total besttime year 0705H = 2009

2BB7D: 07

2BB7E: 05 Month: 0005H = 5

2BB7F: 00

2BB80: 03 ??: 0003H

2BB81: 00

2BB82: 0D Day: 000DH = 13

2BB83: 00

2BB84: 16 Hour: 0016H = 22

2BB85: 00

```

2BB86: 0B    Minute: 000BH = 11
2BB87: 00
2BB88: 2B    Second: 002BH = 43
2BB89: 00
2BB8A: 66    Millisecond??: 0366H
2BB8B: 03
.      .
.      .    Date/Time = 13.05.2009 22:11.43 (German notify)

```

Note that time is GMT, maybe different to local PC-Time ??

A new or reseted track ends here. All bytes of T2 are filled with 00. If the track is used in GenerAlly and a single driver has raced min. 1 round, a Ghost is added to T2.

T24: Ghost

At address Date/Time of Besttimes + $16 \times 7 = 2BB7CH + 70H = 2BBECH$ follows the Ghost. The first two bytes describe the lenght of ghostdata. It is similar to besttime. The Ghost uses 50 frames/sec. Each frame is a set of 12 bytes. These 12 bytes are 2 bytes for each x,y,z-position and rotation x,y,z-axis in GenerAlly-3D-World. Add 1 frame, if besttime not odd.

In object list, the values for x,y are from 0-1024 corresponding to 0-512 in landmap. The values for the ghost in 3D are depending from worldsize and a factor 100/29,5.

I use this factor in my tools and it seems to work fine.

$x (0-512) = x_{hex} / ws / 29,5 * 100$

Address: Value Hex

```

2BBEC: 45    besttime 0745H = 18.61 = 18.61*50 =930,5
2BBED: 07    =931 frames (+1 if not odd)

2BBEE: C0    x-position 2EC0H/ws/29,5*100=11968/140/29,5*100
2BBEF: 2E    x=290
2BBF0: 48    y-position 0448H/ws/29,5*100=26,53
2BBF1: 04    y=27
2BBF2: CF    z-position
2BBF3: 4A
2BBF4: 8F    Rotation x-axis
2BBF5: 08
2BBF6: DD    Rotation y-axis
2BBF7: 25
2BBF8: 01    Rotation z-axis
2BBF9: 8F

```


After the last frame follows a set of 16 bytes. This might be a 128bit hash (MD5,MD4 ?) calculated over the complete trackfile including ghost.

This hash is used in TrackTool - Verifying Ghost

Summary:

0000 05 01 47 52 = 1.05 GR
0004 4Byte pointer to besttimes / lenght of T1
0008 woldsize
0009 waterlevel
0010 Angel
0011 Angel 00/FF +/-
0012 Rotation
0013 Rotation 00/FF +/-
0014 Zoom
0015 Sartline 0/1
0016 4 byte lenght Landmap
0020 Landmap

HMAp (2 byte lenght, always 4096)
Objectlist (2 byte lenght, 1 object = 8 byte)
Pits (1 byte lenght, 1 Pit = 6 byte)
Checkpoints (1 byte lenght, 1 Checkpoint = 8 byte)
AI-Line (2 byte lenght, 1 knot = 4 byte)
Pit-Line (1 byte lenght, 1 knot = 4 byte)
FF FF (2 byte)
Author (30 byte)
Comment (500 byte)
Tracklenght (4 byte)

T2 Besttimes 14 bytes (Besttime + Place 1-6)
Drivernames 7 x 13 bytes
Cars 7 x 1344 bytes
Date/Time of Besttimes 7 x 16 bytes
Gost (2 bytes number of frames, 12 byte each frame)
16 byte Hash

Reset Track:

Load the track file in memory. Go to start of T2

(address offset see byte 4-8 of T1)

Add or change 9625 bytes to 00

Change the first 14 bytes of these 9625 to 70H 17H 70H 17H ...
= 7 x 6000

9625 = 14 byte Besttimes + 7x13 bytes names of drivers
+ 7x1344 bytes Cars + 7x16 bytes Date of Besttimes

Save data length of T1 + 9625 bytes as reseted Trackfile

What is missing?

- Meaning of 2. Byte in Objectlist
- Meaning of byte 5+6 in time/date of Besttimes
- Details of Rotation x,y,z-Axis in Ghost
- Algorithm of Hash

This knowledge about the GenerAlly Trackfile is found by
Try and Error and some experience in Hex-Numbers.

Possible, there are mistakes or other rules.

If you found some or can give supplements please contact:

tom@majortom.de

March 2010 majortom