



**MANU & KIT**

**Путешествие  
сквозь время  
в деталях**

**MANU & KIT**

# **Путешествие сквозь время**

**в деталях**

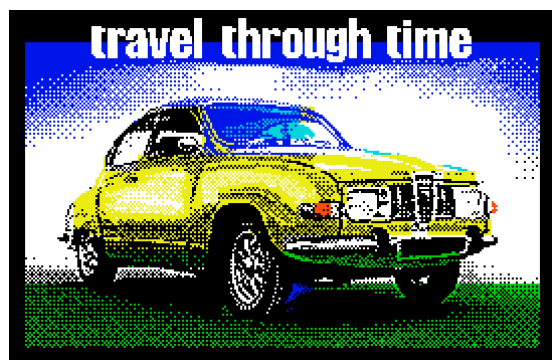
Книга о том, как мы сделали игру Travel Through Time Volume 1: Northern Lights  
для компьютера ZX Spectrum

Создание компьютерных игр—процесс чертовски увлекательный. При работе над крупным проектом всегда возникает огромное количество вопросов и задач, ответы на которые не найти в книгах вроде «Как написать игру на ассемблере». Наша книга не научит вас программировать, но—мы уверены—будет очень интересна всем, кто играл в наши игры и задавался вопросом: «как они это сделали?»

Факты размещены в книге в случайном порядке. Упоминаются и некоторые другие игры, связанные с серией Travel Through Time (далее иногда—ТТТ), а также детали, которые не вошли в финальную версию игры.

## 1.

Изначальный проект игры был полностью посвящен одной известной автомобильной марке. В итоге мы исключили все упоминания названий автомобилей (не будем упоминать их и в книге), чтобы избежать претензий правообладателей, а также добавили другие автомобили и прочие транспортные средства.



Сохранилась первая версия заставки, а также весьма пафосное описание изначального проекта:

«Первая игра в жанре автогонок для ZX Spectrum с фотореалистично изображенной дорогой. Пришло время забыть про полосатую пашню и трековые бордюры, про однообразие уровней и придорожные объекты, вырастающие словно грибы. Трассы в этой игре по-настоящему живые. Лесные массивы и открытые пространства, пригорки, обрывы и стены скал, туннели, перекрестки, функционирующие заправки,

населенные пункты с различными типами зданий, мосты и даже... действующий железнодорожный переезд! А что еще вы не видели в гоночных играх на ZX Spectrum? Предлагаются самые разные виды соревнований: езда на время, дуэль, масс-старт и даже ориентирование на местности, причем на основе реальных карт районов Швеции. И да, в этом виде соревнования перекрестки—действующие. Очки, набранные в гонках, используются для апгрейда автомобиля. Все запчасти в определенной степени влияют на управляемость, разгон и максимальную скорость. Кроме того, отдельная процедура совершенствования реализована для турбины, причем с еще большим количеством параметров. При появлении нового автомобиля турбина переставляется на него, и ее параметры сохраняются».

Как видите, не все изначальные задумки были реализованы (некоторые из них—например, апгрейд—не вписывались в итоговую концепцию), зато появилось много других интересных деталей.

## 2.

В игре используется более дюжины графических форматов и процедур вывода графики—для каждого элемента используются наиболее рациональные методы вывода и хранения данных. Например, вывод машины игрока и вывод столбов на дороге—одна процедура, а вот машины соперников—совсем другая, так как там используются автомаски, попиксельное позиционирование, обрезки, а также масштабирование по вертикали (в небольшой степени). Свои собственные процедуры используются для панорамных планов, «всплывающих» из-за горизонта объектов, деревьев на среднем плане, цветных спрайтов в интерфейсе и кат-сценах, крупных изображений персонажей и так далее.

## 3.

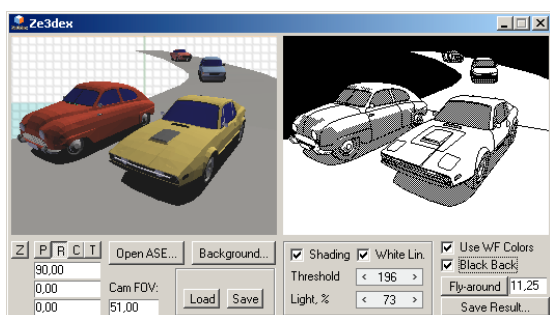
Как и в случае с DRIFT!, все игровые автомобили, а также некоторые машины соперников, были сначала подготовлены в

виде трёхмерных моделей, и на основе этого отрисованы с разных ракурсов. Детализация моделей выполнялась с учетом удобства последующей отрисовки.



#### 4.

При отрисовке автомобилей использовалась собственная утилита для перевода 3D моделей в пиксельную графику. Эту программу под названием Ze3dex мы ранее уже использовали при создании игры DRIFT! Она позволяет отображать в пикселях фактически любую исходную модель, но значительно лучше работает со специально подготовленными моделями с небольшим количеством полигонов и акцентом на необходимые структурные элементы.

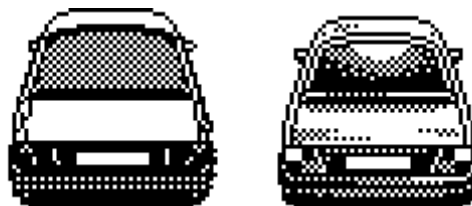


Этот метод фундаментально отличается от типичного случая, когда конвертируется не модель, а её рендер (визуализация). Далее показано, как выглядело бы изображение при типичном конвертировании графики, и как оно выглядит при конвертировании модели.



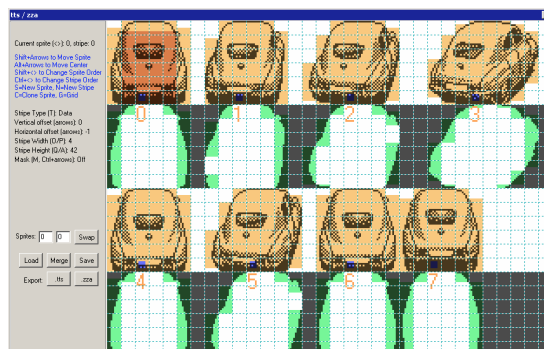
Тем не менее, если крупные изображения, размером с экран ZX Spectrum, с помощью

этой программы получаются изначально приемлемого качества, то такие спрайты, как в Travel Through Time, приходится основательно дорабатывать. В данной игре они фактически перерисованы заново.



#### 5.

Весьма интересен графический формат, использующийся для машин игрока и придорожных объектов. Он изначально разрабатывался для этой игры, потому так и называется: Travel through Time Sprite (.tts). Прежде, чем игра вышла, мы уже использовали его в других играх—DRIFT! и Maureen Miles (известной как Just a gal в выпуске от ZOSYA). Для него был написан редактор спрайтов.



Суть этого формата в том, что спрайты составляются из отдельных частей разного размера, которые могут использовать или не использовать маску. Благодаря этому спрайты сложной формы сохраняются в наиболее компактном виде, а при выводе используется набор процедур, наиболее оптимальных для каждого фрагмента спрайта. Есть подпрограмма для вывода куска шириной 1 знакоместо без маски; то же, но с маской; затем эти две процедуры, но с зеркальным отображением; далее всё то же, но для ширины 2 знакоместа... Выглядит слишком громоздко, зато работает весьма быстро. Есть ограничения: например, в ТТТ ширину фрагмента мы используем равную одному, двум или

четырем знакоместам, а высота всегда кратна двум линиям, как и вертикальное позиционирование спрайта в игре.

Можно также условиться, что мы никогда не используем в игре какой-то вариант фрагмента (скажем, шириной 2 с маской и отзеркаливанием), и тогда можно вкинуть лишние процедуры из кода.

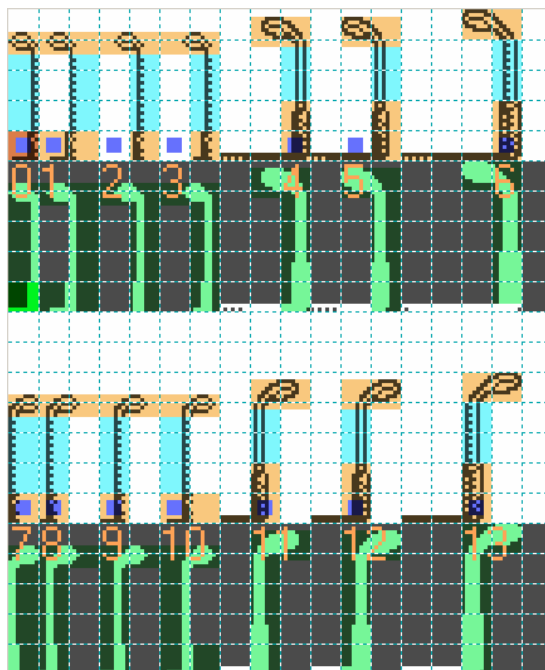
Но это еще не все особенности формата. Он также позволяет сохранять в качестве фрагментов спрайта «столбцы», состоящие из пары повторяющихся байтов—и это не только весьма компактно, но еще и при выводе спрайта высота этого столбца может быть изменена. Как можно догадаться, это идеальный вариант для вывода столбов по обочинам дороги. Более того, для следующих кусков спрайта можно использовать позиционирование по вершине последнего столбца, или же с дополнительным смещением.

Интересный, в общем, формат. Но минусы его очень существенны: позиционирование по горизонтали—только по знакоместам, и сложно организовать обрезку спрайта.

## 6.

Как было сказано выше, объекты по обочинам используют тот же формат спрайта, что и машины игрока. Он не поддерживает горизонтальное смещение «на лету» (надо сказать, процедурный сдвиг—в любом случае крайне плохой вариант для этих объектов из-за медленной скорости вывода), так что смещение прорисовано заранее.

Для самых дальних спрайтов нарисовано 4 варианта горизонтального смещения, далее 2 и единственный вариант для ближнего спрайта. Это обусловлено тем, что горизонтальное движение столбов вдалеке намного медленнее, чем вблизи, и требует более точного позиционирования. Такое решение можно встретить и в других гоночных играх.



Средняя часть столбов выполнена из пары повторяющихся пикселей (принцип описан выше). При выводе спрайта исходная высота корректируется в зависимости от расстояния до объекта.

Можно также заметить, что для некоторых смещенных спрайтов отдельные мелкие детали просто проигнорированы, чтобы не использовать лишнее знакоместо. Тоже своего рода оптимизация—в игре это совершенно незаметно.

Маски используются там, где они необходимы.

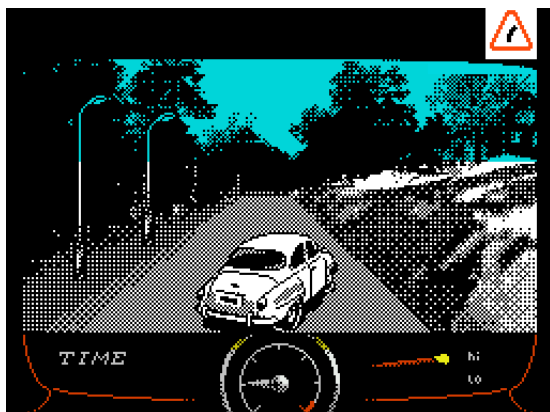
Левый и правый вариант используется для фонарей и знаков направления поворота. Остальные придорожные объекты с левой и правой стороны дороги одинаковы. В играх *Rubinho Cucaracha* и *Travel Unlimited* для фонарных столбов также используется один вариант на обе стороны дороги.

## 7.

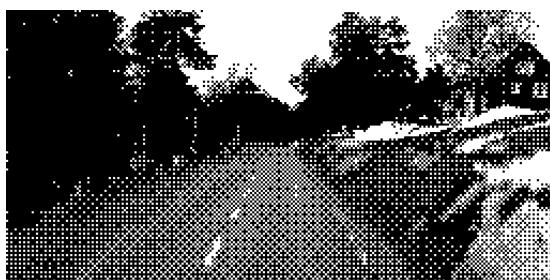
Чтобы всё уместилось в памяти, многие данные сохранены в упакованном виде, распаковываются по мере необходимости. Используется как типовый упаковщик, так и собственные методы упаковки для различных типов данных. Например, для текстов используется система токенов.

## 8.

С данного макета игрового экрана началась разработка игры.



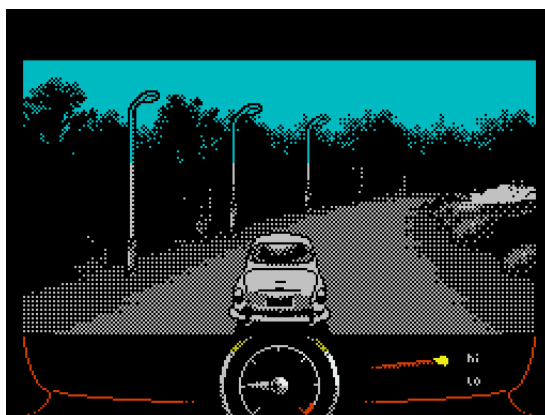
Было отмечено, насколько изображение дороги и окружающих объектов в гоночных играх для ZX Spectrum обычно далеко от реальности и вообще весьма условно. Основой для макета послужила эта наспех конвертированная фотография:



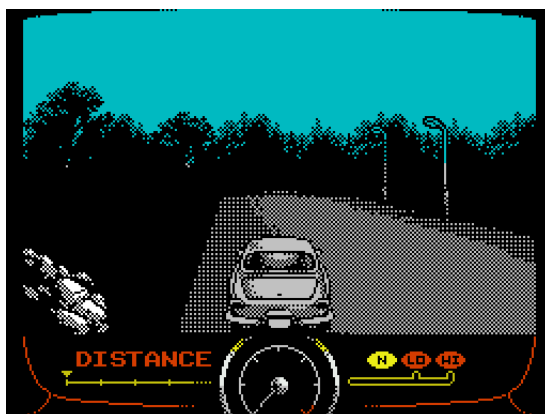
Последующая работа над игрой велась с оглядкой на этот макет, который представлял совершенно иную атмосферу по сравнению со всеми существующими играми. Мы попытались перенести в игру эту атмосферу, учитывая существующие технические ограничения. Например, мы изобразили темные очертания деревьев, медленно проплывающие на среднем плане, довольно «рваную» обочину (в этом возможности нашего движка вообще идеально совпали с исходным рисунком), и в целом картинка в нашей игре—с ее черными заливками—тяготеет к закрытым пространствам, что почти не встречается в гоночных играх для ZX Spectrum. Тем интереснее выглядят время от времени открывающиеся на горизонте панорамы.

Конечно, не всё удалось реализовать в полном соответствии с исходным макетом. Например, холмистую местность за

обочиной с домами и деревьями на ней (всё это в динамике и с сохранением приличной скорости) сделать было бы весьма сложно. Мы вели некоторые работы в области упрощения картинки:



...Ну, или хотя бы каменные валуны в самом начале гонки? Чтобы не рисовать большое количество вариантов разного масштаба.



Увы, в конечном итоге в игру вошла только одна темная скалистая гряда вдоль дороги, которая почему-то представлена в одном единственном 45-м уровне.

## 9.

Итак, в движке TRAVEL, помимо типичных придорожных объектов, используется некий «средний план». Это уникальный элемент, в других играх не встречающийся: объекты вдалеке, которые движутся медленнее, чем придорожные столбы, но всё-таки появляются и уходят за экран, в отличие от постоянного заднего плана. Есть два типа таких объектов: деревья, которые почти сливаются с

задним планом, и «всплывающие» из-за горизонта здания и сооружения.

Еще один уникальный для этой игры элемент, который уже упоминался— панорамные планы. Это набор детально прорисованных изображений, которые иногда появляются из-за горизонта вместо стандартных задних планов. В отличие от последних они, к сожалению, не движутся по горизонтали, но создают бесподобную атмосферу.



Некоторые другие объекты, например, эстакады, под которыми проезжают машины, хотя и не являются совершенно уникальными, но также имеют интересные особенности. Обо всех подобных объектах будет рассказано далее.

## 10.

Для изображения машин соперников (а также в кат-сценах и для других целей) используется простой линейный формат спрайта без масок. Вся прелесть здесь в реализации процедуры вывода спрайта: ведь она использует точное попиксельное позиционирование как по горизонтали, так и по вертикали, автоматическую маску, обрезки по краям экрана и имеет возможность масштабирования спрайта по вертикали. Учитывая весь функционал и скроллинг спрайтов любой ширины «на лету», очевидно, процедура не может быть «мега-быстрой». Тем не менее, она прекрасно справляется с задачами нашего движка. Ведь, как правило, в крупном масштабе мы видим одного соперника, а остальные спрайты, если они есть— намного мельче, и выводятся быстро.

Что касается обреза по краям, они могут быть как автоматическими, так и

принудительными. Это для тех случаев, когда центр машины настолько слева на экране, что уже справа. Но движок при этом знает, что машина всё-таки слева, и резать ее нужно слева.

А еще, при необходимости дискретность горизонтального позиционирования можно настраивать.

## 11.

Масштабирование спрайтов по вертикали сделано так: с каждой линией спрайта увеличивается счётчик, и если он достиг установленного значения, он сбрасывается в ноль, а линия повторяется еще раз. Таким образом, максимальное масштабирование (в два раза) будет выполняться при значении масштаба, равном 1.



При значении, равном 2, спрайт станет полуторной высоты, так как дублироваться будет каждая вторая линия исходного спрайта. В игре используются умеренные значения, более 8. И масштабирование осуществляется только для «ближних», наиболее крупных спрайтов.

Значение -1 (255), или любое другое, большее, чем высота спрайта, выключает масштабирование.

Растяжение спрайта смотрится некрасиво, если дублирующие строки приходятся на участки спрайта с горизонтальными линиями.



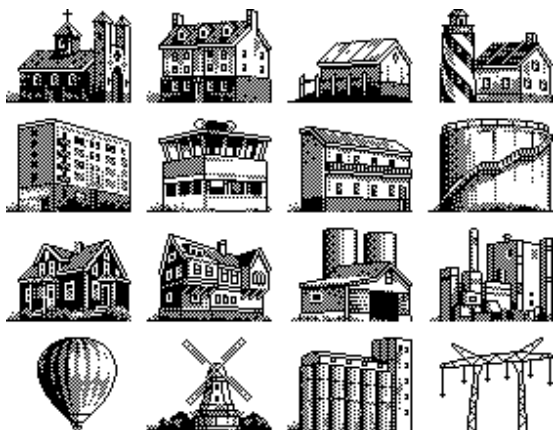
Заранее зная используемые значения масштаба, можно попытаться нарисовать спрайт более удачным образом.

## 12.

В первых рабочих версиях движка этот формат еще не использовался, и грубое перемещение машин соперников на экране (насколько позволял формат .tts) было камнем преткновения на пути дальнейшей разработки. Представьте себе машины, которые уходят со старта «лесенкой». А держать в памяти уже подготовленные смещения для каждого масштаба каждого автомобиля—чрезмерно накладно.

## 13.

Для создания визуального разнообразия в игре используется набор объектов среднего плана, появляющихся из-за горизонта. Всего их 16:



Некоторые встречаются в игре один раз, некоторые повторяются. Индекс объекта указывается в заголовке уровня, если объект присутствует на трассе.

Появление этих объектов всегда происходит на подъеме, когда дорога плавно поворачивает направо, и они, соответственно, всегда движутся влево и вверх, скрываясь в итоге за левым бордюром. И нарисованы они с перспективой, которая подходит для их расположения. Они не масштабируются.

Если бы к началу работы над этими объектами у нас уже использовалась процедура для спрайтов с автомаской и попиксельным позиционированием, то мы наверняка стали бы выводить их с её помощью. Но вместо этого мы поставили себе сверхзадачу реализовать очень

быстрый вывод при плавном перемещении. В итоге для них используется большого размера буфер, где заранее готовятся копии спрайта со смещениями. Более того, для их вывода используются отдельные процедуры, в которых есть возможности необходимых обрезок. Сейчас однозначно можно сказать, что подход был изначально крайне неоптимальный. Даже если вывод спрайта с автомаской и смещением «на лету» медленнее, для такого редкого короткого эпизода это было бы незаметно. Зато немало места можно сэкономить, а возможности вывода лишь возросли бы. Можно было бы рисовать объекты и справа, а еще применять небольшое масштабирование.

В общем, объекты хороши, но процедуры не оптимальны.

## 14.

Изначально стрелку спидометра хотели делать в растровой графике. Передумали, когда прикинули необходимое количество спрайтов.

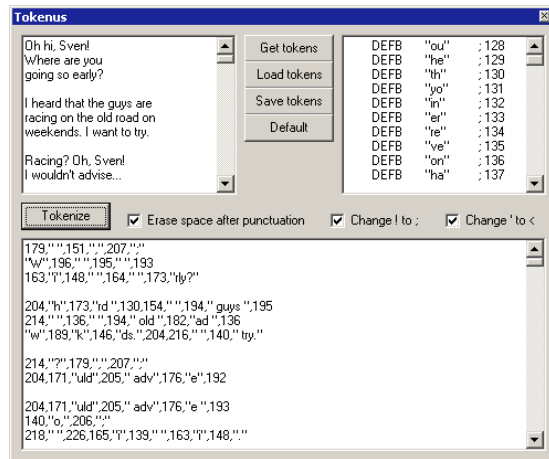


В итоге стрелка рисуется векторной графикой. Для расчета координат отрезков была написана специальная утилита. Полученная таблица вставлена в код программы. Стрелка имеет 62 положения, что более дискретно, чем изменение значения скорости. Она перерисовывается непосредственно на экране, а прежнее положение удаляется. Чтобы меньше тормозить движок, процедура выполняется через кадр, и, конечно, только если положение стрелки изменилось.

Интересно, что координаты внутренней точки, которая изменяется в небольшом диапазоне, записаны в таблице более компактно—одним байтом. Выиграно на этом немного, ведь такую запись нужно ещё интерпретировать в программе.

## 15.

Система токенов, которая используется для упаковки текстов, включает в себя утилиту Tokenus, написанную в процессе создания данной игры.



Она может автоматически определять наиболее часто встречающиеся связи двух символов в исходном тексте (для лучшего результата нужно «скормить» ей максимально полный набор используемых в игре текстов), а также позволяет добавлять другие сочетания букв разной длины. В нашем случае добавлены служебные слова, вроде “the”, “are”, “you”, “here”, “what” и так далее, имена всех персонажей, а также специфические для нашей игры слова, вроде “racing” или “time”. Всего можно задать до 128 сочетаний. (Другие 128 значений байта остаются за буквами, прочими символами и управляющими кодами.) После этого программа переводит исходные тексты в более компактный вид, используя индексы описанных токенов.

Соответственно, парсер в коде нашей игры также должен знать все эти сочетания букв и их индексы, чтобы расшифровать текст. Нужно сказать, что данный метод эффективен только при наличии большого количества текстов и при необходимости прямого доступа к ним. (Ведь парсер и набор токенов занимают немало места!) Но нам как раз это подходит. Типовой упаковщик справился бы с задачей более эффективно, но он потребует распаковки большого блока текста в отдельный буфер, что обычно очень неудобно.

## 16.

Все дорожные знаки в игре, кроме знаков направления поворота, одного типа—предупреждающие. Они состоят из двух спрайтов: пустого щита на столбе и изображения—пиктограммы.



Позиционирование знака по горизонтали более грубое, чем для столбов. Это объясняется необходимостью вывода пиктограммы, а все они представлены в единственном варианте, без смещений и в одном масштабе.

На самой дальней дистанции пиктограмма не выводится (там некое абстрактное изображение), и знак позиционируется с точностью 4 пикселя. Для двух других вариантов масштаба горизонтальное позиционирование познакоместное, и на щите выводится единственный вариант пиктограммы.

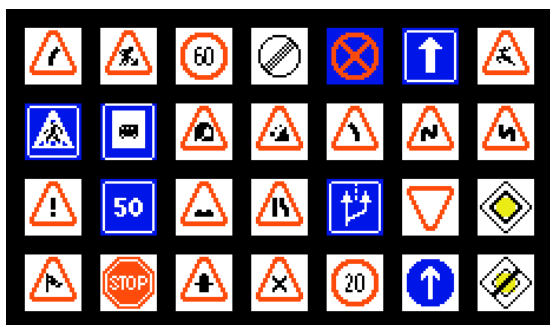
Большинство пиктограмм имеют ширину 1 знакоместо и высоту 8 пикселей, либо меньше. Только знак «Сужение дороги (с обеих сторон)» не уместился в этот размер и использует дополнительное знакоместо с маской.

Четыре последних пиктограммы, начиная с «Прочие опасности», не используются в игре; в спрайтовом наборе их нет. Первоначально была мысль и о косулях, перебегающих дорогу вдалеке (разумеется, не каждый раз после знака, что было бы совсем нелепо), и о перекрестках, но пришлось оставить это для следующих игр серии.

## 17.

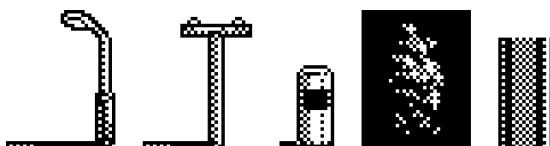
Кстати, еще на первом макете приведено изображение дорожного знака не на самой дороге, а на панели интерфейса. Именно такой приём используется в нашей игре Knights of the Night Roads. И некоторые

пиктограммы переключали в ТТТ именно оттуда:



## 18.

Количество вариантов объектов на обочинах дороги в игре весьма невелико. Помимо двух типов знаков, это: фонари, столбы ЛЭП, маленькие столбики, кустарники и опоры мостов.



Причем мы видим их уже на первых уровнях. Как-то негусто для столь продолжительной игры? На самом деле мы пошли на это осознанно. Во многих гоночных играх восьмибитной эры придорожные объекты меняются от уровня к уровню: ёлки первого уровня меняются на кактусы второго, фон дороги меняется с зеленого на жёлтый, а на горизонте вместо леса вырастают скалы. В *Rubinho Cusagacha*, например, используется именно такая классическая схема.

Но в ТТТ мы решили добиться разнообразия иначе: используя набор объектов среднего плана и уникальных панорам, меняя видимый ландшафт с закрытого на открытый, постепенно добавляя такие элементы, как эстакады, туннели, реки и так далее. Цветовые атрибуты, конечно, тоже разные для разных уровней, а еще есть ночь и зима. И если бы при этом на каких-то уровнях сменился вид фонарей вдоль дороги, это мало повлияло бы на общее разнообразие. Мы посчитали, что вместо этого лучше добавить еще один панорамный план.

## 19.

Спрайты, изображающие кусты, сделаны сразу на черном фоне и без масок, для экономии памяти и увеличения скорости. В игре так рассчитано, что они выводятся исключительно на фоне темной земли или воды. Их нельзя выводить на подъемах, ведь тогда на фоне светлого неба они предстанут черными прямоугольниками с рисунком, вот так:



## 20.

В игре *Crazy Cars II / F40 Pursuit Simulator* на большинстве платформ вдоль дороги изображены типичные столбы, а вот на Atari ST, например, мы видим провода, натянутые между столбами. Подобное мы пробовали сделать и для ТТТ. Ведь процедура рисования линий у нас уже есть, и определение вершин столбов также используется для некоторых целей. Наибольшую проблему представлял бы уход проводов за экран, когда машина минует очередной столб. Велись какие-то ранние работы по реализации этих процедур, но отказались от этого на раннем этапе, так как смотрелось очень некрасиво, исходя из специфики наших столбов. Получалось весьма плоско (в буквальном смысле, ведь вершины столбов на экране находятся на близких высотах для всех расстояний), дёргано, «ну и вообще» — выглядело как дефект изображения. А ведь казалось, что идея интересная.

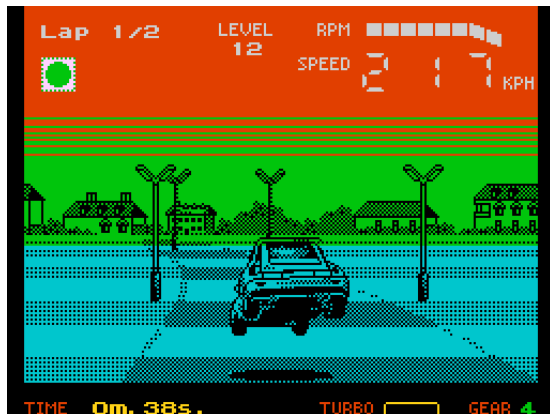
## 21.

Мы долго сомневались в том, делать ли игру одним целым, или разбивать на две отдельно загружаемые части. Очевидный плюс второго подхода: можно было бы уместить больше графики для кат-сцен и фоновых объектов, увеличить количество

кат-сцен, диалогов и уровней, и в то же время одна часть загружалась бы быстрее, чем игра целиком. Отсюда сразу же и минус: обе части вместе занимали бы куда больше, чем 128 килобайт, а значительная часть кода просто дублировалась бы. Мы увидели и еще несколько недостатков деления на две части, связанных с сюжетом игры. Наконец, название «Travel Through Time Volume 1: Northern Lights, Part 1»—это, кажется, уже слишком.

## 22.

В игре нет прыжков (не считая маленьких подпрыгиваний на неровностях), поэтому тень составляет одно целое со спрайтом автомобиля. Она была сделана уже в исходных 3D моделях. В новых версиях движка (например, в спин-оффе Travel Unlimited) появились прыжки, и тень «отделяется» от машины только для тех спрайтов, которые изображают машину в прыжке.



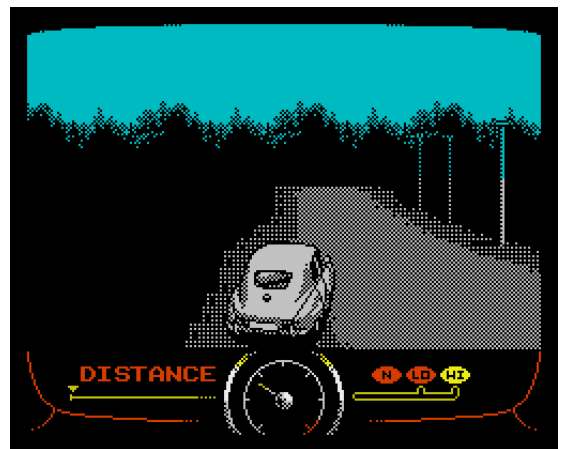
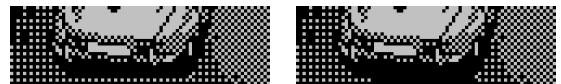
А во второй части основной серии ТТТ прыжки ещё более усовершенствованы: там прыгают также и машины соперников, а неровности на дороге (которые в Travel Unlimited довольно плохо заметны) изображены плавно меняющейся текстурой.

## 23.

На маленькой скорости машина игрока на обочине (и на неровной дороге) раскачивается вперед-назад, а на средних и высоких скоростях начинает подпрыгивать на два пикселя. Вероятно, мы бы сделали

на один, но... вспомните возможности формата .tts.

Чтобы скрыть видимость «приклеенной» к машине тени, мы при подпрыгивании увеличили тень, всего лишь проведя одну короткую чёрную (точнее, залитую пикселями) полосу в фиксированном месте игрового экрана—на второй снизу линии по центру. Почему именно на второй? Потому что текстура первой линии снизу на обочине дороги и так уже черная.



Мелочь, но выглядит лучше. Только вот на подъемах, когда машина перемещается немного вверх, мы уже не используем этот приём.

## 24.

Падающая звезда в открывающей сцене—не спрайт, но процедура.



## 25.

Хочется упомянуть о том, что в эру восьмибитных игр встречалось большое количество вариантов геймплея автогонок, с разными ракурсами и принципами отображения игрового процесса. Даже если не рассматривать вид сверху, сбоку (были и такие) и изометрию, а остановиться на типичном виде из-за машины или из машины, всё равно можно выделить три основных типа:

1) Перспектива не меняется, камера движется вдоль дороги всегда по ее центру, а машина при повороте дороги или руля перемещается по экрану влево и вправо. Например, первая часть *Crazy Cars*, *Turbo Esprit*.

2) Машина всегда находится по центру экрана, камера прикреплена к ней, а перспектива меняется при перемещении машины и камеры поперек дороги (при поворотах); при этом камера не вращается и всегда смотрит вдоль дороги. Это самый распространенный тип, использующийся как для игр с камерой за машиной—*WEC Le Mans*, *Chase H.Q.*, так и с видом из машины, вроде *Chequered Flag*.

3) Полная свобода перемещения. И камера, и машина могут вращаться на 360 градусов. Можно ехать куда угодно. Редкий тип из-за сложности реализации на восьмибитных компьютерах. Например, *Hard Drivin'* или наш *DRIFT!*

Изредка встречается и нечто среднее между этими основными описанными типами. Например, в игре *Crazy Cars II* (которая в целом второго типа) можно ехать почти поперек дороги, и камера поворачивается соответственно.

Так или иначе, для ТТТ мы выбрали самый распространенный второй вариант. Основная сложность его—именно в меняющейся перспективе (имеются в виду воображаемые линии, вдоль которых расположены границы дороги и все объекты), что накладывает ограничения в изображении крупных объектов по обочинам: стены домов, обращенные к дороге, должны соответствовать положению камеры и изгибам дороги.

Можно упростить задачу, помещая дома только на прямых участках, но перспектива боковых стен все равно должна меняться исходя из положения машины. Именно по этой причине в подобных играх редко изображают езду по городу, а если это и сделано, как в *Chase H.Q.*, например,—то показаны лишь дома-«свечки», построенные по принципу столбов и деревьев.

И—вот сейчас обидный спойлер—в ТТТ1 тоже нет никакого города, нет зданий с меняющейся перспективой. Хотя, следующий макет показывает, что некоторые исследования в этой области мы всё же вели.



Идея была в том, что если (на прямых участках) изображать дома довольно далеко от обочины, то, вероятно, можно было бы обойтись без изменения перспективы. В итоге этот вопрос мы оставили для следующих частей серии. («Невозможно же всё уместить в одной игре!»)

## 26.

А на этом скриншоте мы видим дом другого типа, который хорошо смотрелся бы—наоборот—именно в поворотах, без всякого изменения перспективы.



Но и для него, к сожалению, уже не нашлось место в первой игре серии ТТТ.

## 27.

Для всех автомобилей соперников, кроме особенных транспортных средств, используются одни и те же спрайты для двух самых мелких масштабов. К тому же, в этих спрайтах заранее прорисовано смещение для более быстрого вывода и используется иной формат спрайтов—с масками.



## 28.

Все автомобили соперников имеют 5 нарисованных вариантов масштаба (не считая 2 общих дальних). Для ближних вариантов при выводе на экран используется ещё и небольшое масштабирование. Ракурс выбран исходя из того, что обычно машины мы видим не прямо перед собой, а чуть в стороне. Направление спрайта меняется исходя из взаимоположения игрока и соперника, а также поворота дороги.



Некоторые спрайты маши соперников имеют дополнительные фазы для более плавного изменения ракурса:



Причем их количество может быть разным (дополнительные спрайты нарисованы только для определенных дистанций):



## 29.

Некоторые автомобили соперников приходилось дорабатывать исходя из того, как они смотрятся на трассе. Например, этот кабриолет имел три недостатка: выглядел слишком маленьким, был избыточно повернут, а также вызывал вопросы относительно наличия в нем водителя и пассажира (кстати, без пассажира было не обойтись из-за возможности зеркального отражения).



Все недостатки были устранены в итоговой версии, причем последний из них—с помощью «установки» крыши.

## 30.

Один из автомобилей мы попробовали сделать черным, для большего разнообразия. В кадре он смотрелся некрасиво и был исключен. Хотя, возможно, будет использован в одной из следующих игр.



## 31.

Для одного из автомобилей был подготовлен небольшой «пакет модификаций», чтобы сделать из него еще

одну машину, чуть более современную. Из итоговой игры это было исключено: модификации не слишком меняли внешность (определенно, их мало кто заметил бы), но при этом смотрелась машина хуже. К этому времени практически не оставалось памяти, и мы исключали из игры любые спорные детали.



### 32.

Последний из автомобилей соперников, наиболее современный, был полностью перерисован, так как первая версия получилась в неудачном ракурсе. Хотя итоговая версия на первый взгляд не сильно отличается, на дороге она смотрится лучше.



Чтобы подобрать правильный ракурс, мы сделали трехмерную модель. Поскольку машина видна только сзади, для этой модели передняя часть не была сделана.



### 33.

В одном из уровней в качестве игрового транспортного средства используется трактор. Как и для автомобилей, его основные спрайты представлены в формате .tts, однако есть ряд интересных особенностей.



Во-первых, спрайтов намного меньше, чем для автомобилей: представлена только одна степень поворота. Во-вторых, для спрайта, изображающего движение прямо, для экономии памяти хранится только половинка, вторая выводится зеркально. Также это единственное игровое ТС, у которого показано движение колес. Наконец, есть еще передняя часть трактора, которая представлена в ином формате—с автомаской. Это нужно для более плавного перемещения этого элемента. И в последнюю очередь печатается торс Свена.



В геймплее для трактора существенно уменьшена максимальная скорость, что логично. Надо сказать, что на высокой скорости пройти этот уровень, где нужно ехать бок о бок с дядей Бьерном, было бы невозможно.

### 34.

Дым из выхлопной трубы автомобилей и трактора реализован таким набором мелких спрайтов с автомасками:



Пара таких спрайтов движется вверх и постепенно меняет вид от первых спрайтов набора к последним.

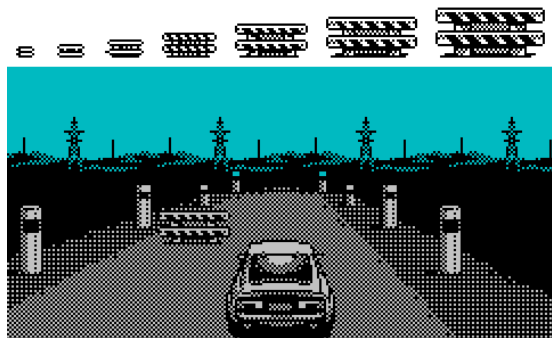
Есть машина, у которой выхлопные трубы с обеих сторон, и для нее дым не выводится дважды, а постоянно меняет своё положение. Смотрится «даже лучше».

Когда машины начинают двигаться, дым исчезает. И только для управляемого Свеном трактора он выводится постоянно. Именно для этого транспортного средства и была изначально предназначена данная процедура. Спрайты дыма даже хранятся в одном наборе со Свеном-трактористом и частью трактора.

### 35.

В техническом плане барьеры на дороге— те же автомобили. Для них отключена большая часть ИИ и звук двигателя, а скорость равна нулю. Процедура проверки столкновений та же самая, и автомобили соперников взаимодействуют с ними так же, как между собой.

Визуально барьеры имеют прозрачные части между планками. На самом деле там изначально нарисована текстура, повторяющая текстуру дороги, поскольку данный тип спрайта использует автомаску, а она не позволяет делать прозрачные части внутри спрайта.



### 36.

Некоторые объекты (барьеры и знаки направления поворота) можно сбивать. (Или можно, но нельзя, как в уровне со слаломом.) Это работает так: когда

столкновение обнаружено и обработано, исходный объект удаляется и запускается анимация, показывающая, как он отлетает в сторону. Она никак не связана с исходным объектом и, фактически, даже не привязана к игровым координатам: столкновение всегда происходит по центру экрана, рядом с машиной, и для анимации отлетающих объектов достаточно каждый раз использовать одну и ту же последовательность экранных координат. Независимо от скорости машины и ее положения на дороге, сбитый знак перемещается в сторону за экран и немного вверх, а сбитый барьер—вверх и вниз.



### 37.

На этом макете изображены следы шин при старте машины соперника. До реализации дело не дошло. В игре есть лишь следы торможения машины игрока.

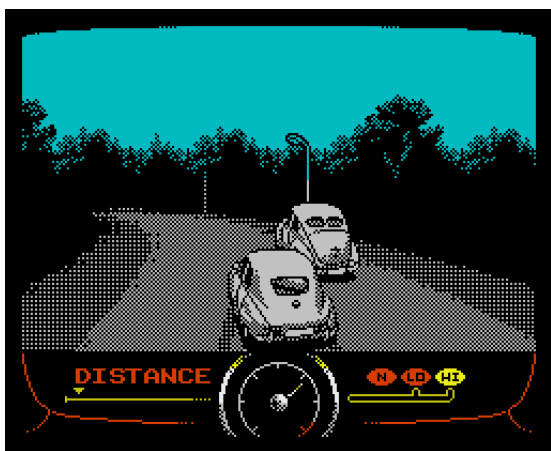


### 38.

Следы при торможении—это процедурный вывод на поверхность дороги вот такого рисунка из двух кадров с фиксированными координатами:



Нет ни изменения перспективы, ни анимации появления и убывания. Но в игре смотрится вполне приемлемо. И, конечно, мы не забыли отключить эту процедуру для мотоциклетного уровня.



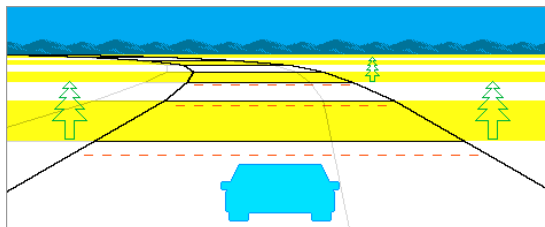
Можно было сделать и спрайтами, как в игре DRIFT!, но процедурный метод компактнее и быстрее.



### 39.

Разберем общие принципы вычисления и отображения дороги на экране для движков подобного типа.

Дорога в карте состоит из сегментов одинаковой длины, для каждого из которых задан ряд свойств, набор которых может быть разным в зависимости от движка: значение поворота, придорожные объекты, ширина дороги, разметка и так далее. В некоторых играх указывается высота (или прирост высоты) для построения честных подъемов и спусков.



Для вывода дороги на экран необходимо знать, помимо текущего сегмента (и, конечно, характеристик всех видимых сегментов), ещё два значения: положение машины на сегменте (то есть пройденное расстояние от его начала; как правило, оно имеет диапазон 0..255) и поперечное положение машины (и камеры) относительно центра дороги. По этим параметрам вычисляется расположение на экране границ дороги и связанных с ними придорожных объектов, строится псевдо-перспектива. Количество отображаемых сегментов обычно небольшое (в TTT1 на экране их восемь). Сегменты всегда перпендикулярны камере, в поворотах они будут лишь смещаться влево-вправо (в отличие от настоящей трёхмерной перспективы, где мы бы заметили их поворот). Кстати, «физические свойства» таких псевдо-поворотов соответствующие: расстояние, проходимое машиной по внутреннему радиусу такое же, как и по внешнему. В продвинутых движках это может быть скорректировано, но это уже отдельная тема.

На представленном схематичном рисунке основные линии показывают примерное отображение дороги для состояния, когда машина находится где-то в начале сегмента и по центру дороги, а впереди виден поворот. В соответствии с перспективой, экранная высота сегментов уменьшается к горизонту, а границы дороги при отсутствии поворотов сходятся (обычно не совсем) к центру горизонта. Поскольку перспектива не реальная, её

правильность зависит только от видения программиста.

При начале движения высота сегментов на экране начинает меняться (пунктирные линии), а вместе с ней и расположение объектов. А при движении машины и камеры поперек дороги (либо игрок рулит, либо машину «выбрасывает» в повороте) меняется перспектива: при приближении к правой обочине правая граница дороги будет становиться более вертикальной, а левая более горизонтальной (бледные линии на картинке).

Процедура расчёта дороги должна учитывать всё описанное выше и на выходе давать как минимум набор горизонтальных координат для левой и правой границы дороги, для каждой отображаемой линии экрана. В случае наличия «честных» спусков и подъемов, когда каждый сегмент имеет еще и высоту (не экранную, а «над уровнем моря»), процедура будет более сложной.

Рисование дороги и объектов на экране при наиболее правильной реализации движка происходит посегментно, начиная с дальнего видимого сегмента. Сначала выводятся текстуры обочины и дороги (текстуры могут меняться от сегмента к сегменту) с бордюрами (если есть), затем придорожные объекты (лучшим вариантом будет привязка объектов к дальней границе сегмента), затем—автомобили соперников. Да, их действительно нужно выводить здесь же, чтобы они правильно перекрывались ближними объектами (так бывает в крутых поворотах) и ближними сегментами дороги (если дорога резко меняет высоту).

Можно добавить процедуру проверки, чтобы не выводить текстуру сегментов, скрытых за ближними сегментами. Однако выводить придорожные объекты и машины в любом случае нужно—они должны торчать из-за холмов.

#### 40.

Даже если описывать вкратце, расчет и вывод дороги—процедуры непростые. Но самое забавное то, что реализация их в

движке ТТТ нарушает многие описанные выше «каноны». Например, дорога выводится не посегментно, а полностью перед выводом объектов, и к тому же снизу вверх, и вообще, выводится не сама дорога... Но об этом будет рассказано позже.

Пока лишь отметим, что под отрисовку дороги отведена одна треть экрана, 64 линии, а расчет границ дороги осуществляется не для каждой линии, а для пары линий, то есть получаем 32 значения для левого бордюра и 32 для правого. Сначала рассчитываются эти значения для сегментов, а затем производится весьма грубая интерполяция для линий между ними.

#### 41.

В ТТТ1 каждый сегмент карты дороги описан одним байтом. Указано только направление поворота (2 старших бита; если оба выключены, то дорога прямая), а остальные 64 значения отданы объектам. Степень поворота, которых всего лишь две, определяется последовательностью значений двух сегментов: 0-1-0-1...—плавный поворот, 1-1-1-1...—крутой.

Изменение ширины дороги и спуски-подъемы не задаются для каждого сегмента, а реализуются специальными процедурами.

#### 42.

При увеличении скорости максимальный угол поворота ограничивается. Подобное можно заметить далеко не только в играх для ZX Spectrum. В то же время, нам просто необходим был дополнительный спрайт машин игрока для движения на очень малой скорости почти поперёк дороги (и в крутых поворотах).



Переход к нему несколько резок (и особенно для показанной выше машины),

но без него было не обойтись. Только управляемый игроком трактор обошелся, но он в принципе упрощен, и для него есть свои особенности визуализации поворота.

#### 43.

Как только в карте встречается знак «Сужение дороги (справа или слева)», тут же запускается счетчик процедуры, которая будет выполнять это сужение. Счетчик увеличивается с каждым пройденным сегментом, и вскоре программа вычисления границ дороги начнет добавлять сужение с указанной стороны.



На этом счётчик не останавливается, и при определенных его значениях дорога таким же образом возвращается к нормальной ширине, затем процедура выключается. Закончить участок сужения можно и принудительно, указав в карте маркер «выключение специальных объектов».

А вот за знаком «Сужение дороги (с обеих сторон)» никакого реального сужения не происходит. Это, скорее, сужение обочины на «сложных участках», как обычно и бывает в реальности.



#### 44.

Спуски и подъёмы—отдельная большая тема. Для начала посмотрим на эти два изображения:



На каком из них спуск, а на каком подъём? Или, может, это вообще сужение дороги? На самом деле, однозначного ответа не существует. Если ориентироваться только на дорогу, то первую картинку мы можем видеть в начале спуска (дорога куда-то ныряет), а затем, когда мы уже спускаемся, видим вторую картинку: внизу спуск заканчивается, дорога возвращается в горизонталь. Аналогично для подъема, только сначала видим вторую картинку (дорога вдалеке пошла вверх), а затем, приближаясь уже к вершине подъема, видим первую: горизонт и весь ландшафт пока скрывается за вершиной.

Соответственно, чтобы точно понять, спуск это или подъем, важно видеть изменение картинки, но ещё важнее—расположение и перемещение линии горизонта и других объектов. Ничто не передает спуски и подъемы так хорошо, как перемещение заднего плана.

Итак, при приближении к подъему дорога вдалеке вместе с линией горизонта начинает очень медленно подниматься вверх, как бы создавая задел для будущего подъема. И при его начале задний план стремительно обрушивается вниз, дорога соответствующим образом выгибается, и теперь мы видим в основном небо. На вершине подъема дорога распрямляется, и задний план возвращается из-за горизонта.

И всё бы хорошо, но как реализовать это в нашем движке, если в нем изначально не заложен функционал для возвышения сегментов? Что ж, не мы первые...

В процедурах расчета границ дороги заложены довольно грубые процедуры, которые при наличии спуска или подъема дополнительно «выгибают» дорогу. Метод крайне «фейковый», но в движке TRAVEL смотрится неплохо, особенно когда на подъеме из-за горизонта всплывают

крупные объекты на среднем плане. Спуски же проработаны чуть хуже и смотрятся не так эффектно.

Кстати, интересно отмечать, играя в различные гоночные игры для ZX Spectrum, насколько по-разному выглядят в них спуски и подъемы. Иногда горизонт остается на месте, а дорога поднимается снизу, как лист бумаги (Full Throttle 2, Beverly Hills Cop). Но это не та реализация, на которую стоит ориентироваться.

#### 45.

Как мы сказали, дорога вместе с линией горизонта иногда поднимается вверх. Но как же нам отобразить её, если мы выводим дорогу только в одной трети экрана? Если рассчитываем её границы только для 32-х пар линий?

Как ни банально, но: нужно рассчитать еще несколько значений и вывести дополнительную часть дороги (помимо восьми отображаемых сегментов) за пределами одной трети экрана. Конечно, возникает ряд вопросов: ведь остальные объекты мы отображаем только для восьми сегментов. Мы поступили интересным образом: для дальних машин добавили некоторую поправку, которая при наличии уходящей вдаль дороги чуть приподнимает экранные координаты машины, как показано на следующем скриншоте (пунктирной линией отмечено окончание основной части дороги). Да, это на самом деле машина из тех восьми отображаемых сегментов, но визуально она находится уже в дополнительных. Очередной «фейк» ради визуальной красоты. А что же по поводу столбов в этой дополнительной части дороги? Всё просто: они не отображаются. Теперь вы это знаете.



Процедуры вывода дополнительной части дороги также тесно связаны с выводом заднего плана, очисткой необходимых областей и рисованием текстуры дороги.

#### 46.

Ещё один нюанс: при перемещениях заднего плана вверх и вниз цветовые атрибуты у горизонта тоже меняются. В игре цвет не формируется в экранном буфере, а всегда выводится прямо на экран. Две части игрового окна в начале уровня окрашиваются определенными цветами, изображающими небо и землю. Соответственно, перекрашиваются в процессе гонки только строки у горизонта. Иногда это заметно. Во многих других гоночных играх поступают так же.

#### 47.

Участки «неровной дороги», предваряемые соответствующим знаком, визуально переданы полосатой текстурой, и машина на них раскачивается вперёд-назад.

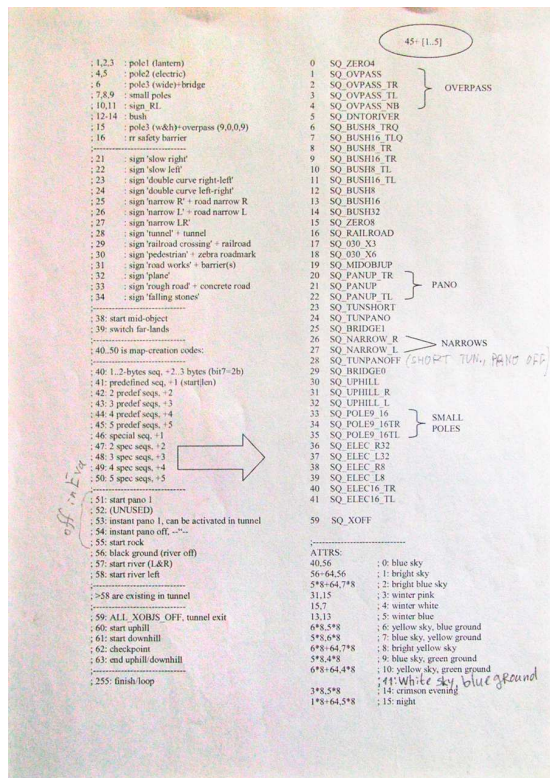


Обратите внимание: на дополнительном участке дороги тоже отображаются полоски, но они (да что ж ты будешь делать!) опять «фейковые». Везде обман, в общем. А вот уже во второй части ТТТ процедура доработана, и полосатая текстура дополнительной части дороги отображается правильнее.

Та же самая текстура используется в туннелях, но там она играет роль ровной дороги.

## 48.

На этом листе представлен полный список объектов, а также предопределенных последовательностей, встречающихся в карте уровня.



В конце листа также представлены используемые в игре атрибуты неба и дороги.

Можно увидеть, что пометок на листе немного. На самом деле, это просто последняя распечатка этого листа, сделанная ближе к концу разработки.

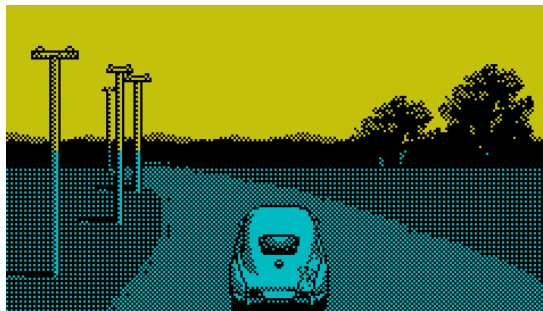
## 49.

Довольно необычен принцип рисования дороги: в игре на самом деле рисуется не сама дорога, а обочины и «земля» за ними. А сама дорога—не что иное, как первоначальная очистка всей трети экрана, заливка её текстурой, либо однородной «шахматкой», либо боле сложный вариант с полосатой дорогой, где от сегмента к сегменту текстура меняется. Есть также возможность честного перехода между этими текстурами.

Без отрисовки обочины дорога выглядит вот так (для наглядности показан переход от однородной текстуры к полосатой):



Затем рисуются обочины, причем в два прохода. На первом рисуется первая снизу линия (слева и справа), затем третья, и так далее через линию. Для этого берутся по 32 заранее рассчитанные значения для левого и правого края дороги.



А затем второй проход: вторая линия снизу и так далее через одну, причем берутся те же самые значения, но делается поправка на ширину обочины, которая убывает к горизонту. То есть, эти линии короче, чем в первом проходе, и в итоге мы получаем изображение и обочины, и земли за ней.



На представленном скриншоте можно заметить, что линии второго прохода помимо того, что они короче, зачастую ещё и начинают выводиться выше, не от самого низа экрана. А, следовательно,

второй проход выполняется значительно быстрее.

Сейчас уже сложно сказать, почему изначально был выбран такой необычный метод отображения дороги, но его используют все игры на движке TRAVEL.

Как уже упоминалось ранее, в наиболее правильной реализации подобных движков сегменты рисуются от горизонта вниз, одновременно со всеми объектами и машинами. В нашем случае направление отрисовки дороги не имеет значения (в отличие от выводящихся позже объектов).

## 50.

На картинке с отсутствующими обочинами хорошо заметен некий графический элемент дороги у горизонта. Дело в том, что наше рассчитанное на треть экрана изображение дороги, состоящее из восьми видимых сегментов, имеет недостаток—слишком резко обрывается у горизонта.

Для исправления этого недостатка над дорогой на горизонте рисуется пара дополнительных отрезков, визуально продолжающих дорогу и следующих за ее поворотами. Эта графика, которую мы условно назвали «дальняя дорога» выводится только при «благоприятных» условиях.

## 51.

Ещё один элемент такого предназначения виден на том же скриншоте: это «дальние столбы». 8 сегментов—это мало. Столбы появляются слишком резко. И мы решили отслеживать содержимое ещё одного сегмента (это быстро, без расчетов границ дороги), чтобы выводить столбы, еще не попавшие в основную область видимости. Это выполняется для двух типов столбов—фонарей и ЛЭП. Причём используется не основная графика столбов, а отдельная весьма быстрая процедура, рисующая точки (инвертирующая пиксели) через линию. Соответственно, на самом дальнем расстоянии столбы двух типов одинаковы.

## 52.

Наверняка вы обращали внимание на «рваное» отображение обочины. Это получается благодаря тому, что при рисовании горизонтальных линий для последнего байта используется не типичная таблица, а со специально добавленным шумом:



«Рваная» обочина на самом деле решает сразу три задачи:

- приближает наше изображение к первому исходному макету экрана, сделанному на основе фотографии;
- скрывает довольно грубо рассчитанную границу дороги;
- добавляет динамику при монотонной текстуре дороги.

Без шума дорога выглядела бы так:



## 53.

Дождь в одной из кат-сцен—вдвое сокращенная процедура из нашей игры Rikki-Tikki-Tavi.

## 54.

Весьма интересны принципы реализации зимних уровней.



Как мы уже знаем, в нашем движке рисуется не сама дорога, а обочины и земля за ними, причем всё изначально рассчитано исключительно на «чёрную» заливку земли (в отличие от последующих игр на этом движке, где земля либо «белая», либо вообще есть возможность обоих вариантов, как в TTT2). Так вот, чтобы изобразить зиму, мы просто использовали инвертированные цветовые атрибуты. Теперь видим светлые деревья на тёмном фоне и так далее. Посмотрите: даже тени от столбов стали белыми на тёмном!

Но как же быть с машинами? Ведь при инверсии атрибутов они должны будут выглядеть вот так:



Чтобы вернуть машинам нормальный вид при инверсии атрибутов, используется инверсия спрайтов. Для машины игрока (описанный выше формат .tts) при инициализации уровня выполняется процедура, которая инвертирует данные спрайта, учитывая маски. Эти изменения обратимы, и при «наступлении весны» спрайт возвращается в исходное состояние (та же процедура выполняется повторно).

Для машин соперников (спрайты которых используют автоматические маски) мы не можем использовать такой метод. Поэтому для них написана процедура вывода с добавлением инверсии «на лету». Это

медленно, но и зимних уровней у нас немного.

Для дыма из выхлопной трубы тоже используется формат с автомасками, и он отображается зимой правильно.

В зимних уровнях также чуть изменено поведение машины, её больше заносит.

## 55.

При тестировании игр есть весьма неприятный момент: отдельные уровни и кат-сцены могут идти правильно, но при попытке пройти всю игру целиком в определенный момент вдруг возникает критическая ошибка, игра зависает и сбрасывается. Причем её сложно отловить, поскольку условия непонятны: скажем, это может произойти на 70-м уровне или после 75-го.

Разумеется, не хватит никакого терпения многократно проходить игру полностью после каких-то изменений, даже с читками. И в TTT мы для этого использовали режим автоматического прохождения. Уровни при этом проходятся весьма условно—для машин отключена проверка столкновений. Такой билд можно запустить в эмуляторе на двадцатикратной скорости и быстро увидеть, дойдёт ли игра до финала.

Однако каждый уровень в отдельности, конечно же, был многократно пройден честно, чтобы найти правильный баланс.

## 56.

Для тестирования Travel Unlimited (в которой, к слову, мы как раз и получили подобную ошибку на старших уровнях), была разработана еще более продвинутая система автоматического прохождения, которая умеет проходить повороты и обгонять соперников. Делает это далеко не идеально, что даже к лучшему: примерно соответствует уровню среднего игрока, не всегда проходя уровень с первого раза. Кроме того, в той игре ещё и автоматически «покупаются» апгрейды, без которых старшие уровни не пройти.

Так что, мы включали эмулятор и смотрели, как игра играет в саму себя.

Еще один весьма полезный режим, использовавшийся при отладке и тестинге: «показывать только кат-сцены».

## 57.

Игра работает только на компьютерах со 128-ю килобайтами памяти. Однако это вызвано в основном большим объемом разнообразных дополнительных объектов и кат-сцен, в то время как самому движку TRAVEL хватит и 48 килобайт. В отличие от игры DRIFT!, здесь используется экранный буфер, а не характерные для 128 К два экрана. Некоторые другие игры на этом движке работают и на 48 К.

## 58.

«В игре нет разметки»,—звучало для нас как укор, но это не совсем справедливо. Мы просто не хотели повторять то, что много раз видели в других играх, и вместо продольной разметки сделали поперечную: пешеходные переходы, линию финиша и пару элементов попроще. И тем, как это выглядит, мы точно можем гордиться.

А что касается продольной разметки, то никаких проблем с ней нет, и мы сделали её уже во второй части серии ТТТ.

## 59.

Для отображения пешеходных переходов и линии финиша используется следующий набор текстур:



Зебра изображена в нём трижды, для разных дистанций, а для линии финиша мы решили обойтись одним вариантом, просто на разном расстоянии рисуем кусок разной высоты.

Принцип вывода горизонтальной разметки завязан на принципе отрисовки дороги,

точнее обочин: разметка рисуется от края до края экрана, а затем обрезается обочинами, и мы видим только кусок на дороге. Если не рисовать обочины, мы увидим такую картинку:



Итак, вопрос обрезки разметки по краям дороги решен. Но как быть с правильностью перспективы? Честно говоря, никак. Разметка рисуется ровно в таком виде, как она представлена в текстурной карте. Вы только посчитайте, например, количество полосок здесь:



...и здесь:



А ведь в динамике всё смотрится просто прекрасно!

## 60.

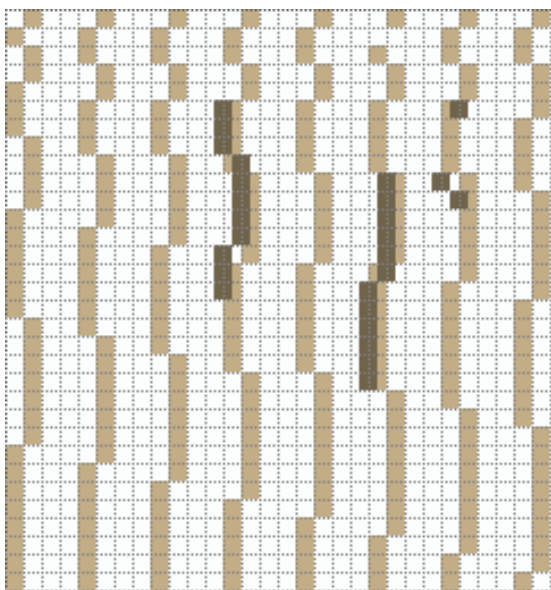
Линии отсечек («checkpoints») и рельсы рисуются подобным образом, но вместо текстуры на всю ширину экрана выводится

один повторяющийся байт. Рельсы должны продолжаться за пределами дороги, поэтому они, в отличие от разметки, выводятся после рисования обочин.



## 61.

На основе этого изображения с клетками разрабатывалась таблица визуальных размеров сегментов дороги (их экранных высот, то есть количества линий).



На экране всегда отображается 8 сегментов. Клетка на этой картинке равна минимально возможной высоте сегмента на экране—двум линиям. 32 клетки по высоте—это 64 линии, то есть треть экрана, в которой и отображается дорога. На рисунке представлены экранные высоты сегментов для 8 кадров. Именно с такой дискретностью мы видим движение на экране. Положение машины на сегменте, которое имеет диапазон 0..255, приводится к значению 0..7. При старте можно заметить, что дорога и

придорожные объекты на низкой скорости движутся несколько грубо, и движение начинается чуть с запозданием после начала набора скорости.

Ближний сегмент имеет высоту от 8 до 1 клетки (от 16-ти до 2-х линий), дальний— всегда минимальный, 2 линии. Сегменты на картинке представлены ломаной фигурой просто для удобства визуального восприятия во время разработки, не более. Тёмные клетки—корректировки для некоторых кадров после тестирования первой версии.

Итоговая таблица (в парах линий) такая:

|         |      |                 |
|---------|------|-----------------|
| RDSEGLN | DEFB | 8,7,5,4,3,2,2,1 |
|         | DEFB | 7,7,6,4,3,2,2,1 |
|         | DEFB | 6,7,6,5,3,2,2,1 |
|         | DEFB | 5,7,7,5,3,2,2,1 |
|         | DEFB | 4,8,6,5,4,2,2,1 |
|         | DEFB | 3,8,6,6,4,2,2,1 |
|         | DEFB | 2,8,7,5,4,3,2,1 |
|         | DEFB | 1,8,7,6,4,3,2,1 |

## 62.

Перед тем, как попасть под поезд, машина свободно проезжает сквозь закрытый шлагбаум. Оказалось, что на фоне такой катастрофы эта деталь вообще не замечается, и мы не стали тратить на это лишние байты и время.

Кстати, первый уровень был намеренно спланирован таким образом, чтобы практически невозможно было избежать контакта с поездом, играя впервые и не зная о нем заранее.

## 63.

В игре встречается множество типов гонок: CHALLENGE, CHECKPOINTS, TIME TRIAL, DUEL, CHASE, JUST DRIVE, SPEED SCORE и SPECIAL EVENT. Вот иконки каждого из них:



На самом деле, типов гонок даже больше, ведь под последним—«особенным»—типом скрывается ряд совершенно разных

соревнований, да еще и с участием уникальных транспортных средств, как игровых, так и трафика. Один из таких уровней—«лесовозы»—имеет даже свою собственную иконку на панели состояния. Чем не отдельный тип?



Некоторые типы гонок встречаются один раз (CHASE), два (JUST DRIVE) или три (SPEED SCORE) раза за всю игру, а CHALLENGE, CHECKPOINTS и TIME TRIAL—это основные виды соревнований.

#### 64.

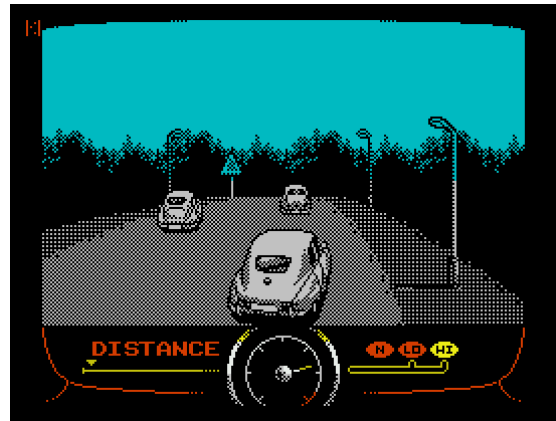
Для каждого вида соревнований описан свой ИИ машин на трассе (соперников, потока, специальных транспортных средств). Поведение соперника в дуэли, безусловно, отличается от такового в гонке, а дуэль на дорогах общего пользования вообще является уникальным соревнованием со своими особенностями ИИ. Но также есть и большой набор процедур, общий для всех машин для большинства случаев.

Если в гонках количество машин на трассе в течение соревнования не меняется, то при езде по дорогам общего пользования машины трафика регулярно обновляются: те, кого давно опередили, ликвидируются, а впереди создаются новые.

Есть также интересная модель поведения под говорящим названием «нельзя обогнать». Она же потом использовалась в Rubinho Cucaracha.

#### 65.

Для того чтобы ИИ машин соперников быстро и эффективно оценивал ситуацию на дороге, мы использовали технологию, которую назвали «ситуационная таблица». Есть даже режим отладки, когда таблица отображается визуально. Вот она, в левом верхнем углу экрана:



В неё вносится примерное расположение машин—как игрока, так и соперников. Таблица имеет малые размеры, благодаря чему ИИ анализирует состояние на трассе быстро и эффективно. Ведь иначе процедурам поведения каждой машины пришлось бы перебирать параметры всех остальных машин соперников, а отдельно еще и машины игрока, и всё это в неудобной форме. Насколько это медленно, даже сложно представить.

Именно благодаря ситуационной таблице соперники могут притормаживать за медленными машинами, выходить на обгон (но при этом не менять полосу, если сбоку от них машина игрока), а также пытаться объехать замедлившегося игрока при появлении сзади. Не всегда получается, но они пытаются.

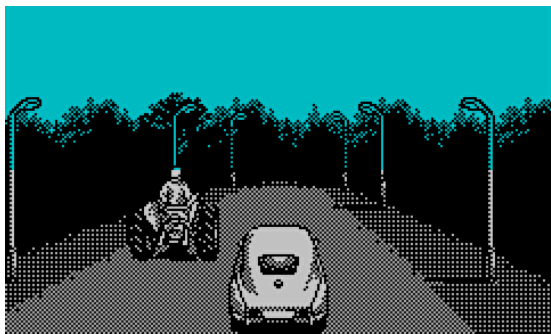
Эта технология сыграла важнейшую роль в геймплее игр на движке TRAVEL. Во многих других весьма продвинутых гоночных играх (не будем показывать пальцем) подобное отсутствует, и машины просто «не видят» ни друг друга, ни игрока.

#### 66.

Скорость машины на подъемах в игре не снижается, она не зависит от ландшафта.

#### 67.

Можно ли на первом уровне подбить трактор дяди Бьёрна?



При приближении к трактору машина Свена автоматически перемещается на противоположную сторону дороги, и попытки взять левее блокируются. Тем не менее, жёсткого ограничения нет, и действительно, можно изловчиться и, заранее прижавшись к левой обочине и разогнавшись, сделать это. Но это ни на что не повлияет. Столкновения на этом уровне разрешены, и как только Свен затем догонит трактор, они с Бьёрном начнут беседу.

## 68.

А что будет, если упустить трактор, либо так и ехать за ним до конца уровня через все эти переезды, без диалога с дядей Бьёрном?

Не получится. С момента появления трактора начинается закольцованная часть трассы. Нас «отпустят» только после диалога. Трактор тоже никуда не уедет, а будет ждать у горизонта.

## 69.

Наконец, ещё один вопрос относительно эпизода с трактором: нужно убрать лишние машины с трассы, если они есть. Сделано это просто: все машины, находящиеся перед нами, резко ускоряются (это очень заметно) и скрываются за горизонтом. А машины позади нас можно просто удалить.

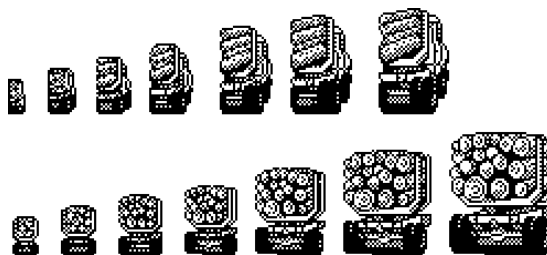
## 70.

Все гонки в игре идут от точки к точке. Нет гонок, состоящих из нескольких

кругов. Эта возможность появилась в следующих версиях движка. Но в ТТТ1 некоторые уровни всё же имеют закольцованную часть, что реализовано упрощенным способом: в месте «склейки» карта имеет одинаковые фрагменты, и в нужный момент машина, а вместе с ней и транспортные средства соперников, переносятся на указанный фрагмент. Помимо первой встречи с дядей Бьёрном, «кольцо» используется на уровне с лесовозами и во время той дуэли с Йоханнесом, где он попадает в аварию. Фактически мы этого даже не замечаем, а этот метод призван лишь закрыть некоторую логическую дыру во время гонки, что наиболее очевидно в эпизоде с трактором.

## 71.

Лесовозы составлены из двух автомобилей, которые едут на фиксированном расстоянии друг от друга, не меняя скорости и полосы. То есть, это не просто два спрайта, а именно две структуры данных для автомобилей. Пришлось также решить несколько вопросов для визуально правильного расположения частей на экране и их обрезки при уходе лесовоза за экран. Задняя часть учитывает расположение передней, и используется ограничение по максимальному горизонтальному расстоянию между ними на экране.



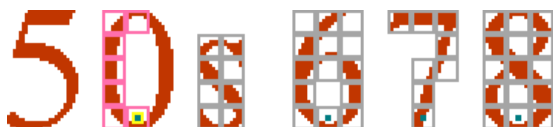
Передняя часть отдельно от задней выглядит забавно узкой.

## 72.

Игра разделена на 4 десятилетия—с 50-х по 80-е, большая часть сюжета приходится на 50-е и 60-е.

### 73.

Перед каждым десятилетием выводится заставка с автомобилями той эпохи и крупной надписью: «50s / 60s / 70s / 80s». Эти символы, что естественно, выполнены обычными спрайтами. Для экономии от цифры «0» хранится половинка (2/3, если точнее), а цифра «5» записана отдельно, так как выводится в начале игры один раз, и далее эта область памяти используется для видеобуфера.



Также шрифт дополнительно упакован. На упаковке выиграли немногим более 100 байтов, но в крупных проектах любые оптимизации важны—главное, чтобы они не вредили.

### 74.

Панорамные планы зачастую логически связаны с другими объектами. Например, на уровне 45 нам открывается красивый вид на скалистый берег и маяк.



Затем—после короткого туннеля—панорама меняется на открытое море...



...И тут же на подъеме перед нами из-за горизонта появляется тот самый маяк, уже вблизи.



### 75.

Этот уровень (который, к слову, встречается после уже примерно полутора часов спидрана) вообще отличается детальной проработкой карты с многократной сменой панорам и большим количеством других деталей. Например, начинается гонка у знака «Падение камней».



Вскоре после этого дорога проходит мимо скалистой гряды.



Для её рисования использованы отдельные процедуры, благодаря которым скалы следуют за изгибами дороги, а формат графики тот же, что и для некоторых крупных элементов интерфейса—с простой упаковкой по столбцам, что

The figure consists of two line graphs, one above the other, both showing the percentage of respondents who believe the U.S. should take more action to reduce global warming. The x-axis for both graphs represents time in years, and the y-axis represents the percentage of respondents.

**Top Graph (2006-2008):**

- Y-axis:** Labeled "Percentage of respondents" with a scale from 0 to 100 in increments of 20.
- X-axis:** Labeled "Year" with markers for 2006, 2007, and 2008.
- Legend:**
  - U.S. should take more action to reduce global warming (solid line)
  - U.S. should take no more action to reduce global warming (dashed line)
- Data Points (approximate):**

| Year | U.S. should take more action (%) | U.S. should take no more action (%) |
|------|----------------------------------|-------------------------------------|
| 2006 | 65                               | 35                                  |
| 2007 | 75                               | 25                                  |
| 2008 | 80                               | 20                                  |

**Bottom Graph (2009-2011):**

- Y-axis:** Labeled "Percentage of respondents" with a scale from 0 to 100 in increments of 20.
- X-axis:** Labeled "Year" with markers for 2009, 2010, and 2011.
- Legend:**
  - U.S. should take more action to reduce global warming (solid line)
  - U.S. should take no more action to reduce global warming (dashed line)
- Data Points (approximate):**

| Year | U.S. should take more action (%) | U.S. should take no more action (%) |
|------|----------------------------------|-------------------------------------|
| 2009 | 75                               | 25                                  |
| 2010 | 80                               | 20                                  |
| 2011 | 85                               | 15                                  |

```
@ABCDEFGHIJKLMNPQRSTUVWXYZ-./0123456789:!'";?
abcdefghijklmnopqrstuvwxyz
```

**77.**

Некоторые панорамы были нарисованы на основе фотографий, другие—полностью от руки. Например, на этом скриншоте

**GO!**

TIME: 01.59    SPEED: 100    GEAR: LO

Единственная деталь: при проезде под эстакадами и в туннелях верхнюю «фейковую» часть игровой области

приходится временно заливать черным цветом, но это весьма несложно.

Такая же тема с визуальным расширением экрана используется и во многих последующих играх на движке TRAVEL—Rubinho Cucaracha (где изображены неподвижные облака), Travel Unlimited (где полосатый градиент имеет некоторую динамику на спусках и подъемах) и всех частях основной серии ТТТ.

## 80.

В некоторых уровнях по сюжету главный герой не может оказаться на подиуме, либо должен сделать это, но не может выиграть гонку. Это реализовано забавнейшим образом: машины, которые должны прийти впереди, уже стоят на финише! Ведь на старте мы в любом случае не можем видеть всю колонну.

## 81.

Всего в игре используется 9 видов панорамных планов, «всплывающих» из-за горизонта. Наиболее красивые и детально проработанные из них мы, конечно, поспешили показать уже на первых уровнях. В дальнейшем они становились намного проще, вроде этого:



Но это не потому, что нам было лень, а для экономии памяти. Ведь они хранятся в упакованном формате, и если первая панорама занимает почти килобайт, то самая простая—менее полутора сотен байтов. И лишь одну из наиболее детальных панорам мы приберегли для уже упомянутого уровня 45.

## 82.

Технически вывод панорам устроен очень любопытно: как уже было сказано, они хранятся в упакованном виде (специально созданный формат), но распаковываются не как графические данные, а в виде уже

готовой программы для максимально быстрого вывода этой графики на экран—она «вшита» в код. Разумеется, для распаковки нужен отдельный буфер. Для этой цели используется область памяти, где ранее хранились данные для поезда (который мы больше не встретим в игре). Эта область практически совпадает по размеру с наибольшей из распакованных панорам.

## 83.

На уровне 55 мы видим наиболее простой из панорамных планов—взлётное поле. Но оно «украшено» взлетающим самолетом.



Помня о традиции связи объектов, мы вскоре видим поднимающуюся из-за горизонта вышку управления аэропорта.



Ну а в самом начале уровня мы уже видели знак «Низколетающие самолёты», и теперь убедились, что он стоял там не просто так.



84.

Кстати, вышка управления срисована со старой (ныне не существующей) вышки аэропорта финского города Оулу. Ведь действие игры на этих уровнях вслед за Эвой перенеслось в Финляндию.

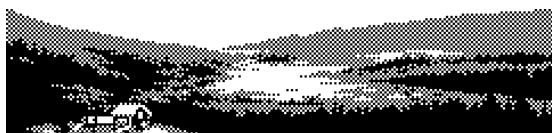


85.

Как уже упоминалось, некоторые панорамные планы намеренно упрощались для экономии памяти. Например, эта панорама...



...изначально выглядела так:



86.

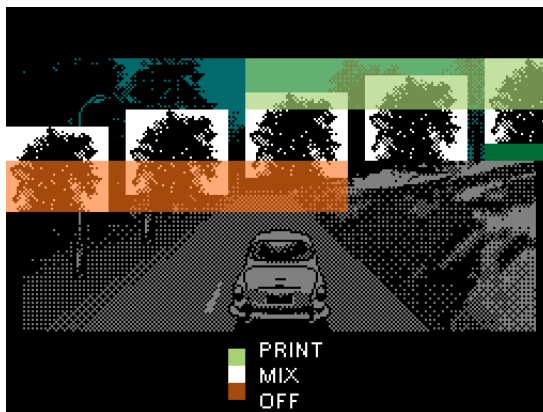
Помимо взлетающего самолета, на одной из панорам планировали изобразить движущийся корабль. Не очень понравилось, как он смотрится, и в итоговую версию игры он не вошел.



87.

На следующем макете велись поисковые работы по отображению деревьев на среднем плане: какая часть должна выводиться напрямую, какая смешиваться с задним планом, а какую можно и не выводить при данном расположении

деревя. И всё это—чтобы добиться и достаточного соответствия исходному макету, и сохранить высокую скорость.



Деревья на среднем плане—не только наиболее уникальный объект в игре, не встречающийся больше ни в одной гоночной игре на ZX Spectrum. Он ещё и был весьма сложен в реализации. Нужно было учесть множество аспектов при перемещении деревьев и выводе их на экран. Они должны не только плавно двигаться, но и в некоторой степени следовать за изгибами дороги (при этом не должно возникать ощущения их движения вспять), а также «дружить» со всеми изменениями ландшафта и заднего плана (чтобы, например, на подъеме не оказались оторванными от земли). Поскольку ландшафт в игре быстро меняется с закрытого на открытый, деревья среднего плана должны вовремя уходить с экрана. Их перемещение фактически происходит по собственным законам, не связанным напрямую с текущей игровой ситуацией, а скорее адаптирующимся под неё.



И вот еще интересная деталь: хотя кажется, что деревья в движении увеличиваются, эта иллюзия достигается лишь их постепенным появлением (движением вверх) на фоне заднего плана. В действительности, используется одно единственное дерево одного масштаба.

А вот горизонтальное смещение, как видите, прорисовано заранее. Причем, копии с разным смещением несколько отличаются друг от друга. В движении это

выглядит весьма интересно—добавляется некий динамический шум.

Разумеется, для деревьев используются собственные процедуры вывода, учитывающие все описанные выше особенности и позволяющие плавно уводить их за экран. Одновременно на среднем плане может существовать ровно два дерева—либо по одну сторону дороги, либо по разные.

## 88.

Нарисовать шлагбаум было не так просто, как может показаться: ведь он на разной дистанции до камеры может находиться еще и в разных состояниях: закрытом, открытом и переходном (открывающемся или закрывающемся). При этом, поскольку он расположен горизонтально или диагонально, а не вертикально, наши процедуры «растяжения» спрайтов для него не подходят. А еще, будучи объектом редко встречающимся, он не должен занимать слишком много места в памяти. В итоге составили его из отдельных кусков, количество повторений которых зависит от расстояния до объекта. Перекладина имеет всего два варианта толщины. При проезде на малой скорости смотрится не очень красиво, но в целом задача решена.



## 89.

В логике игры прописано, что каждый второй шлагбаум на уровне закрыт в ожидании приближающегося поезда. Однако лишь на первом уровне есть два железнодорожных переезда. Начиная со второго уровня область памяти, занимаемая поездом, используется под

другие цели (распаковка панорамных планов). Соответственно, поезд уже невозможен, но игра никогда и не возвращается к первому уровню.

## 90.

В игре используется ручное переключение передач и «двухступенчатая» коробка, так характерная для аркадных автогонок. Есть также нейтральное положение. Первая передача включается автоматически при начале движения.

В некоторых других играх на этом движке (Rubinho Cucaracha и Travel Unlimited) переключение передач автоматическое, а кнопка джойстика активирует турбину.

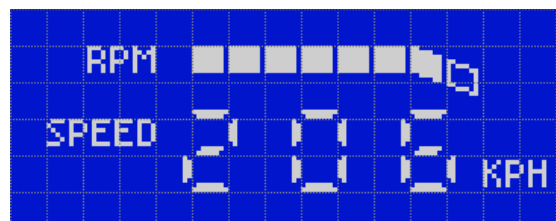
## 91.

На панели состояния текущая передача показана изменением атрибутов—«подсветкой» соответствующей иконки.

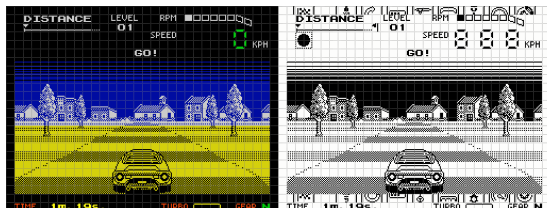


Это, пожалуй, наиболее быстрый и экономный способ отображения пользовательской информации в интерфейсе, и он часто применяется в играх на ZX Spectrum исходя из особенностей экранной области памяти этого компьютера.

Тем же самым способом часть графики интерфейса может скрываться полностью: для этого используются цветовые атрибуты с одинаковым значением бумаги и чернил. Именно так устроен цифровой спидометр в игре Travel Unlimited: на экране заранее нарисованное значение «888» скрывается за атрибутами.



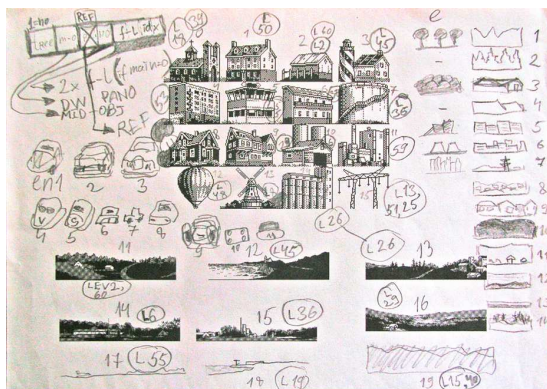
Больше того, в этой игре ради экономии памяти ещё и все элементы графики «гаража» (иконки апгрейда) скрыты в двух визуально пустых строках экрана.



Так можно поступать, если нигде в игре не предполагается полная очистка экрана. А иногда за атрибутами скрывается даже часть кода. Мы делали так, например, в игре Don Quixote.

## 92.

Весьма любопытный лист, на котором прорабатывалось использование панорам, машин, дальних планов и «всплывающих» объектов на разных уровнях.

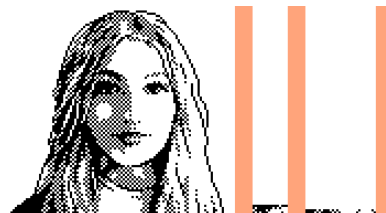


Также в левом верхнем углу можно увидеть проект формата одного байта из заголовка уровня, где в отдельных битах описываются некоторые параметры трека или наличие таких параметров (чтобы следом считывать определенные значения, если только они присутствуют). Подобный компактный метод записи важен для нас, поскольку уровней у нас очень много.

## 93.

Все портреты персонажей были нарисованы на основе случайных фотографий из интернета. Для спрайтов используется формат с простой упаковкой

по столбцам, который дополнительно упакован еще и типовым упаковщиком.



Наибольшее количество мимики у Свена и Ирмы (но набор отличается). А Йоханнес и дядя Бьёрн умеют только рот открывать.

Ирма—единственная, кто умеет смотреть в сторону: на Мари, сидящую рядом с ней, и на Свена во время поездки.

## 94.

Имена персонажей выбирались на основе данных об именах новорожденных в Швеции в конкретные годы. Например, если Свену в середине 50-х около 20 лет, то он родился в середине 30-х.

Действительно, это имя было тогда на втором месте по популярности. И для остальных персонажей как минимум выполняется правило: «да, его могли так звать».

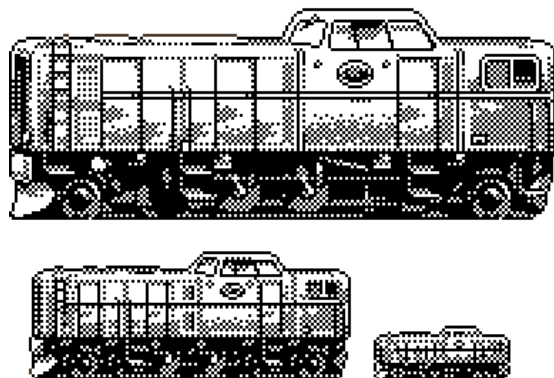
## 95.

Локомотив в первом уровне не вымышленный, это изображение реально выпускавшейся модели SJ Тр, которая использовалась на шведских железных дорогах. В 1953—1954 годах было выпущено 25 единиц, так что можно предположить, что действие игры начинается не ранее 1953 года.

Этот же тепловоз играет весьма драматичную эпизодическую роль в фильме «Вторая жизнь Уве», в котором, к слову, также значительное место уделяется автомобилям.

## 96.

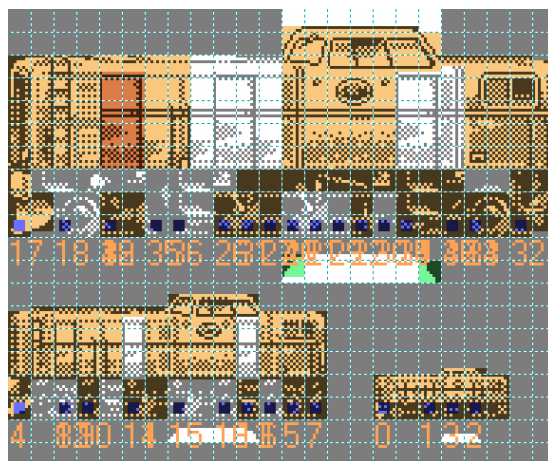
Сколько нужно вариантов разного масштаба для спрайта поезда? Если столько же, сколько и для автомобилей, то, учитывая размеры локомотива, он занял бы очень много места в памяти. К счастью, его вполне можно отображать более дискретно: ведь если мы успели остановиться перед переездом, то поезд уже не будет приближаться и увеличиваться, а если не успели—стало быть, скорость машины велика, и поезд тоже будет приближаться очень быстро; резкое изменение масштаба (всего три варианта) не бросится в глаза.



Кстати, мы упомянули логику появления поезда: программа определяет, что мы едем слишком быстро, и выпускает поезд в нужный момент. Если мы не начали торможение заблаговременно, то уже поздно это делать при появлении поезда. А если визуально машина уже успела проскочить переезд, поезд все равно продолжит двигаться в нижней линии экрана, и избежать столкновения не получится. Всё это происходит так быстро, что на детали и условности не обращаешь внимания. И только в том случае, если мы едем достаточно медленно и успеваем остановиться перед закрытым шлагбаумом, поезд появляется не мгновенно, а через некоторое время.

## 97.

Хотя локомотив имеет всего три варианта масштаба, он все равно занимал бы немало места в памяти при таких размерах. Логично, что в игре он составлен из набора мелких спрайтов, многие из которых повторяются несколько раз. Это также решает проблему обрезки изображения по краям экрана, так как именно для этого типа спрайта обрезка не предусмотрена.



## 98.

Было подготовлено изображение машины, разбитой при столкновении с поездом. В игре не используется, так как было почти незаметно в столь маленьком эпизоде. Вместо этого поезд быстро уносит за экран обычный спрайт машины.

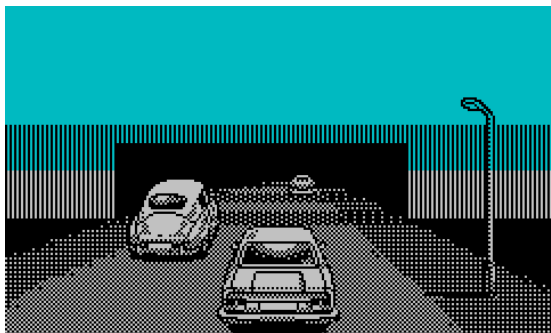


## 99.

Еще одна связанная с поездом деталь, на которую мало кто обратил внимание: после диалога с Дядей Бьёрном до самого финиша на дороге нет других машин. Конечно же, это потому, что их взаимодействие с поездом не предусмотрено. А также были бы проблемы с последовательностью вывода спрайтов, так как поезд выводится вне процедур сортировки объектов (именно поэтому сразу за переездом нет столбов на обочине).

## 100.

Туннель—один из наиболее интересных объектов игры. Его реализация использует большой набор процедур. Наиболее эффектно выглядят въезд в туннель...



...и выезд из него.



Для этого используются процедуры заливки, рисующие стены и перекрытия с позиционированием по краям дороги. Важнейшая деталь—обязательная смена текстуры дороги на «полосатую». Ведь в самом туннеле нет никаких объектов, и движение передается исключительно текстурой. Такая же текстура иногда используется и на некоторых других участках дороги.



Почему мы не добавили фонари, как в Chase H.Q.? Например, чтобы не повторять эту игру во всём, а также чтобы немного сэкономить память и время, затраченное на разработку... Хотя, в созданной позже второй части игры мы улучшили туннели, в первую очередь добавив именно фонари. Также в TTT2 машина стала, наконец, задевать стены.



В туннелях, разумеется, не выводится задний план, а вместо него выполняется сплошная заливка. Вывод всех объектов снова включается перед самым выездом из туннеля.

## 101.

Также туннели могут использоваться для смены заднего плана, в том числе панорамы могут не только «всплывать» из-за горизонта, но и включаться и выключаться моментально, будучи замаскированы коротким отрезком туннеля.

Можно еще упомянуть о том, что перед каждым туннелем стоит соответствующий дорожный знак. Он ставится автоматически, когда в карте встречается идентификатор «начало туннеля».



## 102.

Первый уровень игры—единственный, где кат-сцена встречается по ходу уровня, разбивая его на две части. Эта сцена одна из наиболее детально проработанных.



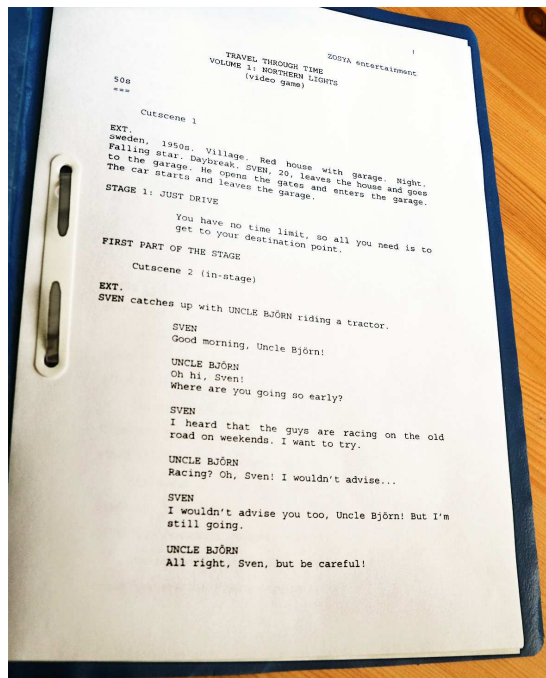
Помимо спрайтов, анимация здесь передается с помощью скроллинга машины вверх-вниз и еще более интересного скроллинга деревьев—не только вправо, но и чуть вниз. Та же процедура используется в сцене поездки с Ирмой. А сама графика деревьев—это графика заднего плана уровня.

Только вот Свен не похож сам на себя.

## 103.

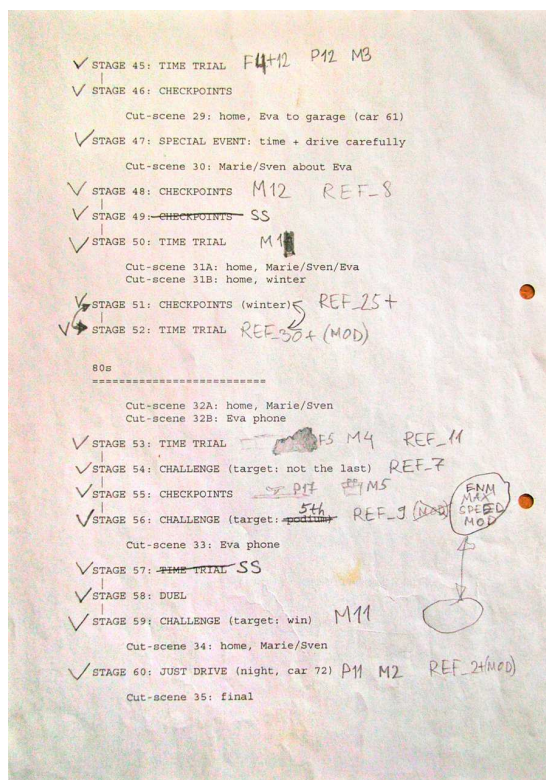
Сценарий игры, написанный в формате киносценария («американка»), содержит 21 страницу и 36 сцен. Продолжительность кат-сцен в игре—около 17 минут. Это

действительно близко к классической схеме «1 страница сценария—1 минута экранного времени».



## 104.

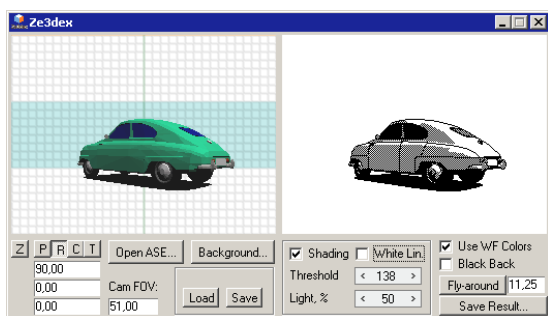
Перед написанием полного сценария был составлен список уровней и кат-сцен игры с кратким описанием некоторых основных событий.



На этом листе показана часть списка с большим количеством пометок.

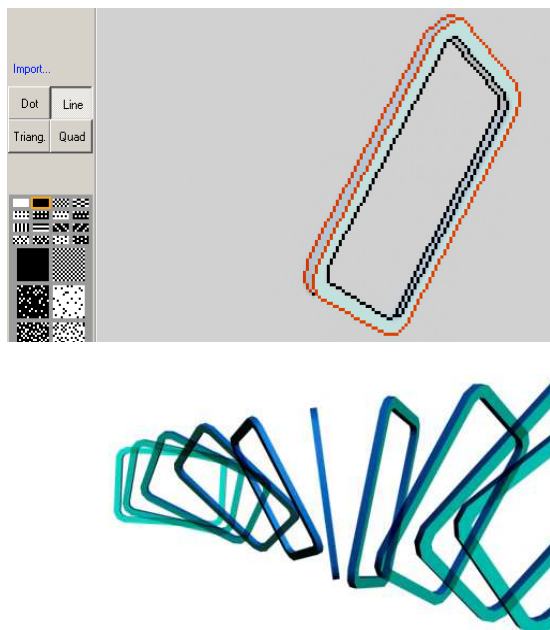
## 105.

Изображения автомобилей на экранах «с годами», конечно же, тоже были сделаны с помощью утилиты Ze3dex, но в отличие от графики геймплея, они дорабатывались в минимальной степени, так как для крупных картинок результат получается приемлемым.



## 106.

Анимация рамки, вылетающей «из-за экрана», была выполнена в 3D, и затем с помощью наспех написанной утилиты вручную перерисована в векторный формат—координаты точек для вывода отрезков.



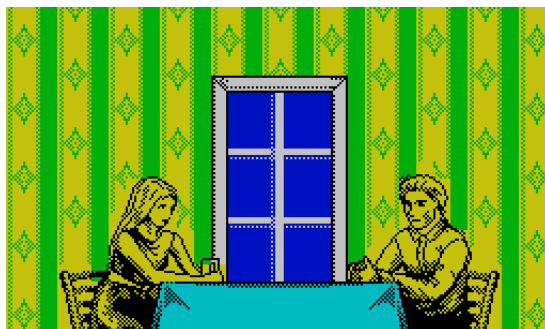
В игре та же самая процедура вывода линий используется и для стрелки спидометра. Получилось 10 кадров векторной графики, а в финальном кадре для рамки используется уже растровая графика.

## 107.

Для игры Rubinho Cucaracha движок был безжалостно упрощен до минимального функционала. Впоследствии при создании ТТТ2 за основу был взят именно упрощенный движок, на который заново «накидывались» туннели, эстакады и прочие выброшенные ранее объекты. Кому-то это покажется нелогичным, но кодеры знают, как сложно работать с «перегруженным» движком, в котором процедуры раскиданы «по щелям в банках».

## 108.

Рисунок обоев на стене в доме Свена художник скопировал с обоев в доме, где он находился в момент создания этого изображения.



## 109.

Игра состоит из 60 уровней, есть также 20 бонусных уровней, которые становятся доступны после полного прохождения и повторной загрузки игры (в конце основной игры нам рассказывают, как их запустить).

## 110.

Бонусные уровни оформлены в виде скрытой игры с собственным названием «Eva's Race». В них мы играем за Эву, как и в 80-е годы основной игры.



На самом деле, трассы бонусных уровней в основном повторяют трассы основной игры с небольшими отличиями: в них отсутствуют панорамные планы и объекты, появляющиеся из-за горизонта, а также используется другой набор задних планов. Вот соответствие трасс бонусных и основных уровней:

|     |   |    |   |    |    |   |   |   |    |    |
|-----|---|----|---|----|----|---|---|---|----|----|
| Бон | 1 | 2  | 3 | 4  | 5  | 6 | 7 | 8 | 9  | 10 |
| Осн | 6 | 53 | 3 | 54 | 46 | 2 | 7 | 4 | 51 | 58 |

|     |    |    |    |    |    |    |    |    |    |    |
|-----|----|----|----|----|----|----|----|----|----|----|
| Бон | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| Осн | 50 | 30 | 12 | 8  | 34 | 19 | 24 | 10 | 13 | 57 |

Бонусные уровни в основном проходятся чуть легче, чем соответствующие им основные, поскольку используются исключительно автомобили последних лет, более быстрые.

В целом эта игра в игре занимает самый минимум дополнительного места в памяти. Даже что касается её крупной картинки в меню, она изначально хранится в той области памяти, которая будет

использоваться в качестве экранного буфера. А затем, для дальнейшего отображения, копируется в область, которая в бонусной игре не используется.

То есть, максимум, что мы могли бы выиграть, не делая эту бонусную игру— это чуть больше места для интро основной игры (например, для картинки в меню). Никакие дополнительные кат-сцены все равно уже не уместились бы.

## 111.

Шрифт в бонусной игре немного изменен. Зачем? Потому что можем, наверное. Для этого используется набор верхних частей цифр и заглавных букв.

0123456789: ! ' " # \$ % & ' ( ) \* + , - . / : ; < = > ?  
 a b c d e f g h i j k l m n o p q r s t u v w x y z  
 0123456789  
 a b c d e f g h i j k l m n o p q r s t u v w x y z  
 0123456789  
 a b c d e f g h i j k l m n o p q r s t u v w x y z

## 112.

Более того, для Eva's Race хотели ещё и всю панель интерфейса изменить. Но сочли это совсем уж бессмысленным.



## 113.

По логике, в бонусных уровнях движение должно быть правосторонним: действие, видимо, происходит в Финляндии, да и Швеция этому времени уже перешла на правостороннее движение. Однако мы видим дорожные знаки слева. Всё просто: это фактическая ошибка. Первые 38 уровней основной игры используют левостороннее движение, а в бонусных уровнях отсчет начинается заново—и движение снова левостороннее. Если бы бонусных уровней было не 20, а более 38, мы бы увидели смену стороны движения, как и в основной игре. Ошибка не была своевременно замечена.

## 114.

Задний план в игре представлен изображениями, повторяющимися по горизонтали и смещающимися влево и вправо с разной скоростью в зависимости от поворота. Ширина картинок всего 64 пикселя (8 знакомест), а максимальная высота 24 пикселя. Да, скроллинг выполняется быстро, картинок в памяти уместается много, но всё-таки нельзя не согласиться, что размер очень маленький. Во многих других гоночных играх, наших и не наших, используется задний план шириной в экран, не говоря уже об убийственных четырёхэкранных (1024 пикселя) фонах игры DRIFT! Смириться с таким маленьким размером помогает только то, что объекты среднего плана, смешиваясь с задником, добавляют разнообразия и динамики. Ну и, знание того, что в Chase H.Q. куски фона лишь немногим шире, конечно же.

Всего для TTT1 нарисовано 19 задних планов. Некоторые используются только в бонусной игре.



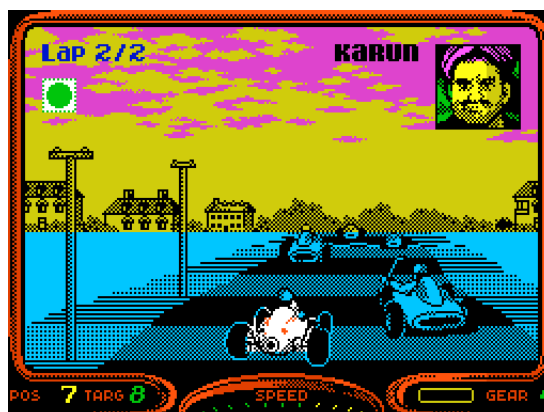
Высота и наполнение задних планов решительным образом влияет на ощущение камерности уровня или, напротив, открытого пространства.

## 115.

Работа над игровым движком была начата в 2013-м году. Хотя проект был на долгое время заморожен (и затем переосмыслен), в основе игры содержится тот самый код отрисовки дороги. На момент создания игр DRIFT! и Maureen Miles с полностью

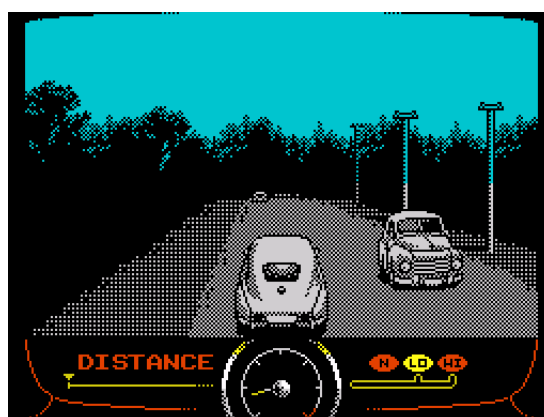
другими движками, у нас уже было понимание, что движок TRAVEL должен стать основным для наших гоночных игр. И, хотя в разработке есть более простые проекты на этом движке, мы стремились сначала непременно выпустить основную игру будущей серии TTT. В планах выпустить еще несколько игр этой серии, а также другие гоночные игры, в том числе и для 48K.

Одна из таких игр—Rubinho Cucaracha.



## 116.

В игре планировались встречные машины на некоторых уровнях, но они выглядели чрезмерно быстрыми даже на скорости, близкой к нулю, поэтому от них решили отказаться. Был подготовлен набор спрайтов для одного такого автомобиля. Как вариант, была идея сделать проезд встречного автомобиля в самом начале игры, когда автомобиль игрока стоит у левой обочины. Эта деталь еще раз подчеркнула бы тот факт, что в то время движение в Швеции было левосторонним.



Однако мы сочли нерациональным хранить полноценный набор графики автомобиля для столь маленького эпизода.



Можно пофантазировать на тему того, как интересно смотрелся бы встречный трафик на уровне с лесовозами. Увы, в этом случае выход на обгон превратился бы в лотерею.

## 117.

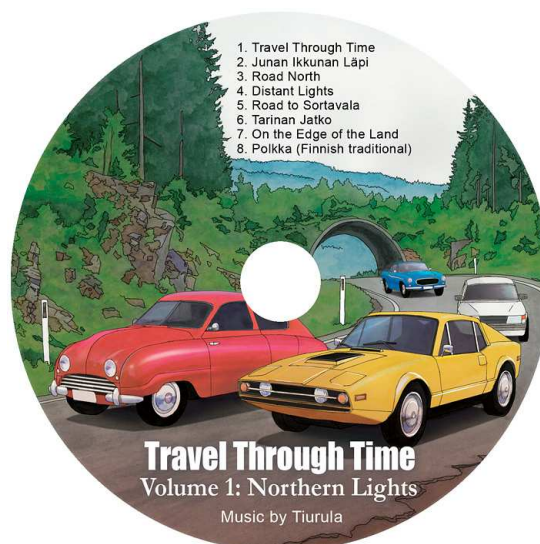
Музыка в игре—АУ-аранжировки композиций группы Tiurula с двух альбомов (одноименном Tiurula и Tie Pohjoiseen): в игре звучат треки Junan Ikkunan Läpi, Road to Sortavala, Night in Hiitola и небольшой отрывок из Ensilumi для озвучивания «неудачи». А основная тема была написана специально для игры.



На прилагаемом к игре компакт-диске представлены именно инструментальные версии в исполнении Tiurula. Во время игры можно—на выбор—слушать АУ или CD.

Список композиций на компакт-диске отличается от используемых в игре:

1. Travel Through Time
2. Junan Ikkunan Läpi
3. Road North
4. Distant Lights
5. Road to Sortavala
6. Tarinan Jatko
7. On the Edge of the Land
8. Polkka (Finnish traditional)



## 118.

Почему в игре не используется музыка во время гонки? Прежде всего, это существенно замедлило бы игровой процесс. Дело не столько в самом плеере, сколько в адаптации некоторых процедур к работе с прерываниями. Так что вместо этого мы поработали над звуком двигателей—для автомобилей как игрока, так и соперников. Да, нам известно о гоночных играх с музыкой во время гонки, но всё-таки скорость куда важнее. К тому же, музыки в игре и без того предостаточно.

## 119.

Некоторые технические подробности: для увеличения скорости игра активно использует метод копирования и вывода через стек (кто знаком с ассемблером, это конструкции POP—PUSH или LD—PUSH) с запретом прерываний. Именно эти методы «плохо дружат» с музыкальным плеером. Там, где использование стека нерационально, используются цепочки LDI, что, хотя и значительно медленнее, чем копирование через стек, но быстрее, чем инструкция LDIR (которую в «уважающих себя» играх используют только вне основного цикла).

Для вывода текстуры обочин, а также других сплошных заливок, наиболее быстрым вариантом оказалась цепочка

инструкций LD (HL),r: INC L с расчетом точки входа.

В кат-сценах, где играет музыка, вместо быстрого копирования экранного буфера пришлось использовать более медленные методы, но там это и не критично.

## 120.

В одном эпизоде игры описывается переход Швеции на правостороннее движение—так называемый Dagen H, который произошел 3 сентября 1967 года. Когда мы еще рассматривали разделение игры на две части, и были менее стеснены в объеме данных, мы подумывали над тем, чтобы сделать короткую игровую сцену на улицах Стокгольма, где не нужно было делать ничего, кроме как медленно перепарковать машину на другую сторону дороги. Поскольку дальнейшее движение по городу не планировалось, можно было бы схитрить: довольно детально и крупно нарисовать здания, находящиеся вблизи, и при этом как можно меньше связываться с масштабированием объектов. Хотя, полностью отказаться от этого в рамках нашего движка всё равно не получилось бы—для зданий необходимо было бы проработать процедуры динамики, связанные не только с масштабированием, но и с изменением перспективы.

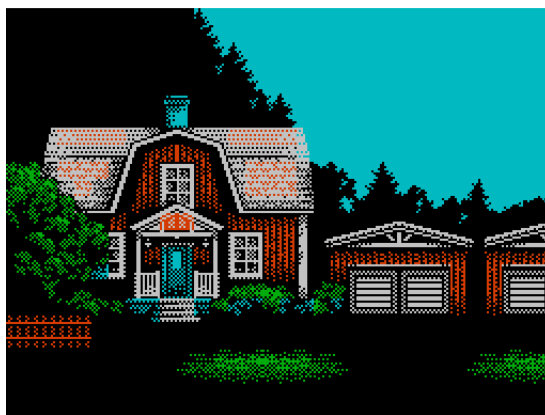
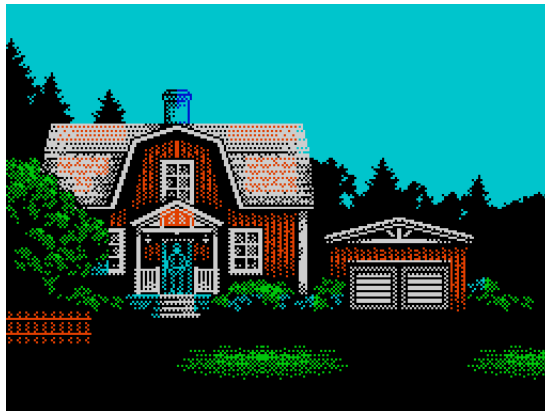
Это в любом случае заняло бы много места в памяти, но можно было бы поступить так же, как с поездом в первом уровне: поставить этот эпизод в начало планируемого второго блока игры, а в дальнейшем не возвращаться к нему и использовать освободившуюся память для других целей.

В итоге мы отказались от такой сцены, а это важное событие вплели в сюжет игры совершенно иным образом, получив наиболее драматичный и душевный эпизод.

## 121.

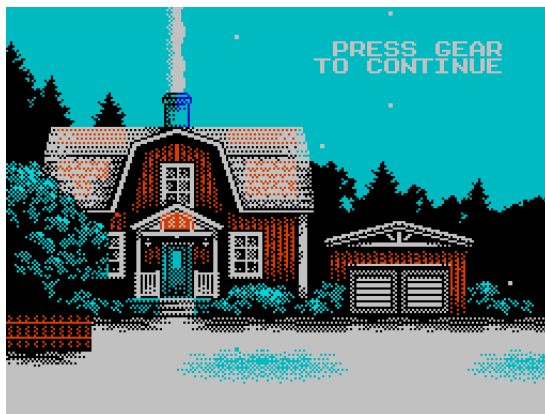
Наверняка немногие заметили, но деревья за домом Свена с течением времени

выросли. Это о нашем внимании к деталям. Также появился второй гараж (что, конечно, более заметно).



## 122.

Зимняя картинка с домом Свена почти полностью состоит из спрайтов летнего состояния, которые были перекрашены. Дополнительную атмосферу придает падающий снег, дым из трубы и музыкальная композиция «Night in Hiitola» группы Tiurula.



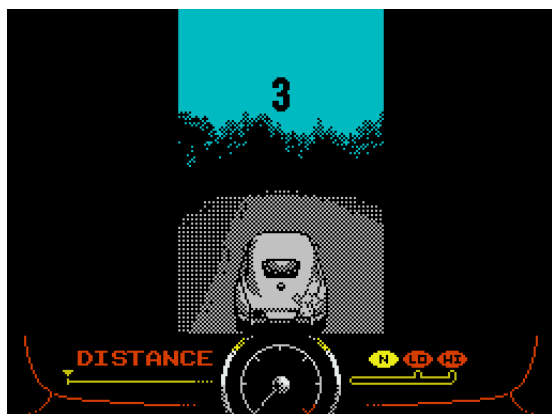
А еще есть та же сцена ночью (перекраска и несколько изменённых спрайтов), осенью с желтыми деревьями и ранней осенью—всего несколько «пожелтевших» атрибутов.

### 123.

Частота кадров в игре—около 25 FPS при отсутствии машин соперников вблизи. Это выше, чем во многих существующих гоночных играх с подобным наполнением экрана.

### 124.

В начале уровня мы видим эффект вертикальных «шторок»—игровая область постепенно появляется от центра экрана. Это реализовано атрибутами. В начале уровня экран окрашен нулевыми атрибутами («чёрное на чёрном»), а шторки постепенно окрашивают его в цвета, заданные для неба и земли.



Есть и закрывающиеся шторки, которые закрашивают экран чёрным. В этот момент запрещаются другие процедуры, связанные с атрибутами, вроде их изменения на горизонте.

### 125.

Проверка столкновений с машинами соперников (точнее сказать, каждой из машин трафика с машиной игрока) происходит довольно просто: если машина соперника на экране находится ниже определенного порога (задан константой),

проверяется горизонтальное расстояние до центра машины игрока. Если оно меньше еще одного порогового значения, происходит столкновение. Большинство машин схожих размеров, поэтому используют один и тот же хитбокс. И только для управляемого игроком мотоцикла в процедуру проверок вносятся модификации.

### 126.

При расчете дороги, позиционировании придорожных объектов и машин соперника движок использует сглаживание данных:

$$\text{ЗНАЧЕНИЕ} = (\text{РАССЧИТАННОЕ} \\ \text{ЗНАЧЕНИЕ} + \text{ПРЕЖНЕЕ ЗНАЧЕНИЕ}) / 2$$

Для еще более плавных перемещений это выполняется дважды. Главный минус сглаживания—запаздывание перемещения объектов, что в игре весьма заметно по движению столбов относительно обочины. Однако без сглаживания нам было не обойтись: исходные вычисления слишком грубы, и перемещения были бы дёргаными. Придорожные объекты вместо плавного ухода в сторону, дёргались бы влево-вправо.

Отставание перемещения более заметно на высоких скоростях. Чтобы нивелировать этот негативный эффект, для придорожных объектов на высоких скоростях делается лишь одинарное сглаживание движения, в то время как на низких—двойное.

Перед началом гонки производится многократный расчёт попавших на экран придорожных объектов, иначе мы бы увидели работу сглаживания: как столбы плавно встают на свои места. Впрочем, напрашивается и иная реализация—временное отключение сглаживания.

### 127.

Интересно, что благодаря сглаживанию движения само по себе реализовалось изображение смены полос машинами соперника. Дело в том, что для упрощения

данных, мы изначально решили хранить для машин трафика не точную горизонтальную позицию на дороге, а лишь маркер нахождения на левой либо правой её стороне. И когда понадобилось добавить возможность смены полос (для чего мы просто одномоментно меняем маркер полосы), оказалось, что машины на экране уже умеют менять полосу плавно. Хотя и слишком быстро, но это в целом соответствует их представлению в структуре данных, где, как сказано, есть только «справа» или «слева».

## 128.

Для расчета горизонтальной координаты машин соперника на экране берутся значения левой и правой границы дороги для той линии экрана, на которой находится машина. Находится середина дороги, а затем, в зависимости от того, на какой из двух полос находится машина,— середина этой полосы. Координата сглаживается на основе предыдущего значения.

Интересно, что при сужении дороги получается, что машины сами по себе подруливают, хотя никаких процедур для этого написано не было.

Но вот незадача: середину дороги и каждой из полос мы не всегда можем определить точно: ведь если дорога уходит за экран, то оставшиеся значения равны границе экрана: при расчете мы не выходим за границы значений байта. Получается, что граница дорога как бы резко уходит вниз на экране, а учитывая перспективу, это равно резкому сужению. И машина, вместо горизонтального приближения к границе экрана, резко начнет двигаться вниз.

Никаких изящных решений придумано не было. Для исправления ситуации мы находим место «обрезки» дороги, чтобы затем неким образом корректировать положение машин. Ещё ближние машины дополнительно смещаются в сторону в зависимости от вертикальной позиции на экране (выполняется для 32-х нижних линий игрового экрана).

## 129.

Весьма милая деталь: в том эпизоде, где мы впервые видим, что Мари и Свен встречаются, над домом одна за другой падают две звезды, а не одна, как в открывающей сцене. Вот этот момент в сценарии:

Cutscene 14

EXT.

Sven's home. There's a SECOND GARAGE next to the first. Night. TWO falling stars.

INT.

Sven's home. SVEN and MARIE.

Забавно, что до этого в сюжете они ни разу не разговаривали друг с другом.

## 130.

Уровень с погоней—очевидная дань игре Chase H.Q. Жаль, что он только один раз вошел в сюжет (и то с натяжкой).

Подобно играм серии Chase H.Q., мы изобразили и картинку задержания преступника. (В первой части Chase H.Q. на ZX Spectrum этого не было, там были только портреты преступников.)

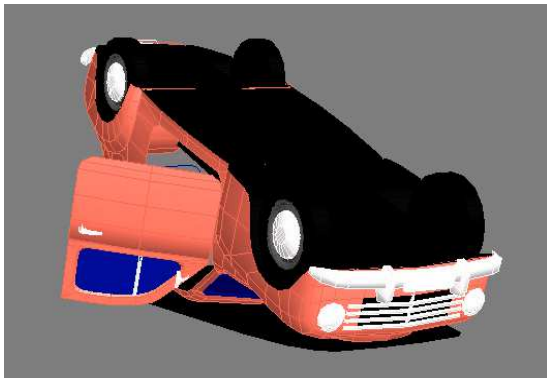


Обошлись геймплейной графикой и таким небольшим набором спрайтов:



### 131.

Перевернутая машина была нарисована на основе визуализации трехмерной модели.



Как и в случае с поездом, нам потребовалось не так много спрайтов разного масштаба, тем более что в логике игры прописано, что мы не подъезжаем близко к перевернувшейся машине.

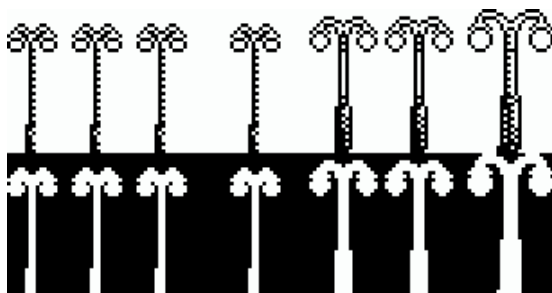


### 132.

Игра использует практически всю память компьютера. В каждом из доступных участков памяти осталось не более 20 байтов.

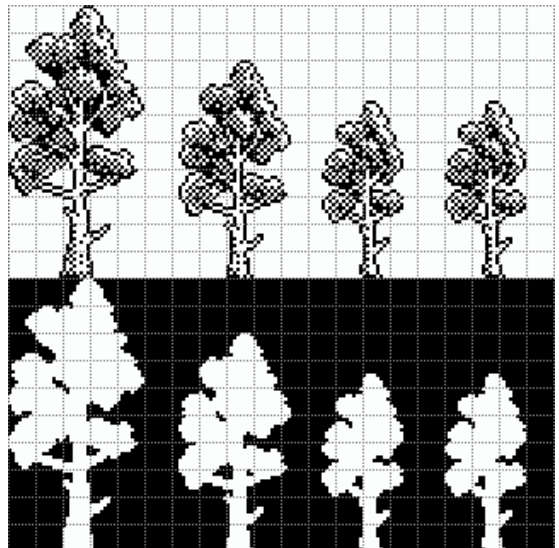
### 133.

Фонарный столб, который мы позже использовали в Rubinho Cucaracha, был нарисован во время работы над TTT1. Файл так и называется: pole\_other\_game.



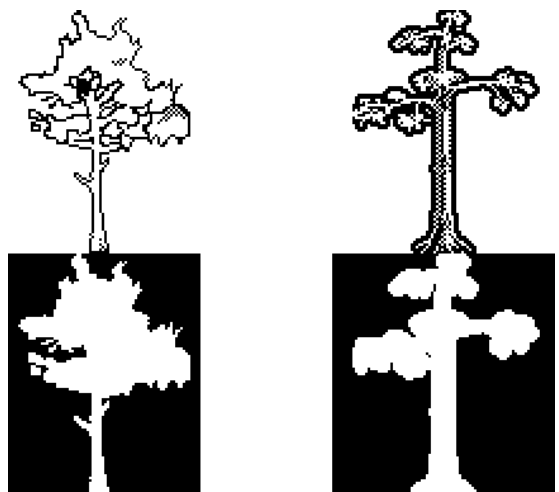
### 134.

Деревья в игре нарисованы на среднем и дальнем планах, а вот по обочинам встречаются лишь небольшие кустарники. Тем не менее, в процессе работы было нарисовано и протестировано несколько вариантов деревьев.



Для данного дерева заранее прорисовано смещение только для дальнего состояния, а вообще, вариантов разного масштаба явно недостаточно.

Главные проблемы использования деревьев в данном движке: для красивого плавного увеличения размера нужно много вариантов масштаба, что занимает большое количество памяти; кроме того, большой размер спрайта существенно замедляет скорость игры и появляется вопрос обрезки спрайтов по краям экрана, что, например, для тонких столбов не так существенно.

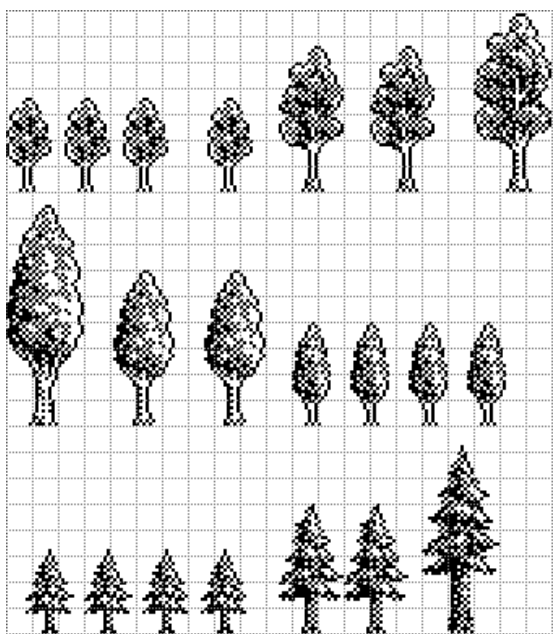


Последнее недорисованное страшненькое дерево использовалось прежде всего для тестирования растяжения отдельных частей спрайта. Также мы пробовали использовать формат спрайтов с плавным растяжением, автомаской и обрезками, как для машин соперников, но в итоге решили отказаться от деревьев по обочинам дороги.



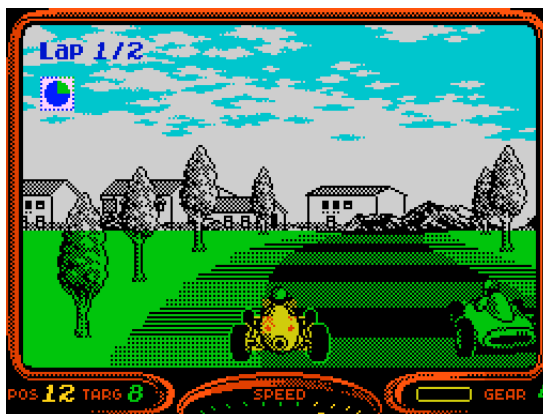
### 135.

А вот в более простой и совсем несерьёзной игре Rubinho Cusaracha, где мы не предъявляли особых требований к реализму, использовано сразу три вида деревьев. При этом с некоторыми изменениями использованы наработки из игр TTT1 и DRIFT!



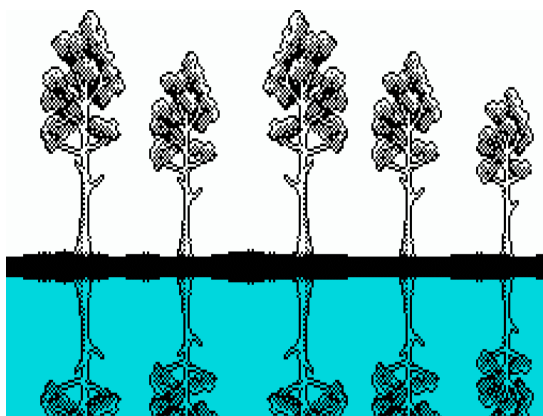
Обратите внимание: для самых дальних спрайтов нарисовано 4 варианта горизонтального смещения, далее 2 и единственный вариант для ближнего спрайта. Так же, как и для столбов в TTT.

Так они выглядят в игре:



### 136.

Следующая картинка, которая просится в какую-нибудь приключенческую игру, также появилась в процессе «работы над деревьями». Здесь определенно не хватает фигуры рыцаря на коне.



### 137.

Главный минус спрайтов с автоматической маской—это возможность применения только для объектов с довольно простым замкнутым выпуклым контуром. (Исходя из принципов построения автомаски, выпуклость контура должна соблюдаться для горизонтальных сегментов.) А вот, скажем, для автомобилей с открытыми колёсами между колесом и кузовом будет стираться часть фона.

В TTT1 все спрайты машин соперника имеют выпуклый контур, за исключением совсем уж мелких незначительных деталей. А вот при создании Rubinho

Cucaracha пришлось подумать, как подружить автомаску и открытые колеса. В том числе и поэтому машины соперников имеют более широкий кузов, чем машина главного героя (которая, как и в других играх на движке TRAVEL, использует иной формат—.tts с масками).

В итоге некоторые «дыры» в спрайтах соперников Рубиньо всё же оставлены, что можно заметить на следующей картинке, но они весьма незначительны.

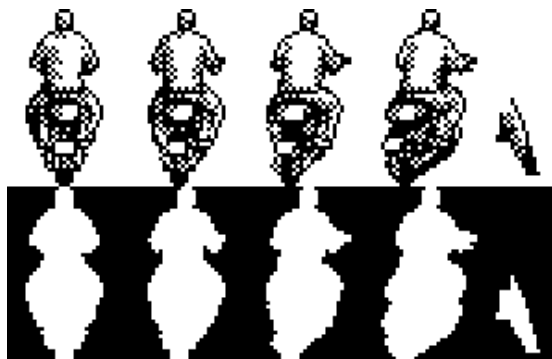


### 138.

На скорости близкой к нулю Свен-мотоциклист выставляет ногу.



Это спрайт, который просто выводится поверх основного, причем хорошо смотрится при любом повороте мотоцикла.



Для мотоциклетного уровня мы вообще обошлись очень компактным набором спрайтов.

### 139.

Предусмотренная возможность смены текстуры земли по ходу гонки используется для изображения воды: иногда дорога пересекает реку или проходит рядом с водоемом.



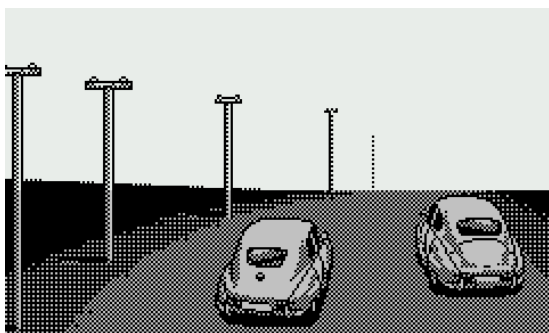
Исходя из ограничений движка, текстура в любом случае должна быть темной, но небольшие барашки волн на ней выглядят нормально. Кроме того, для них сделана анимация, передающая движение воды (между выводом линий выполняется процедура смещения текстуры).

В дополнение, чтобы вода выглядела чуть реалистичнее, у горизонта добавлены отрезки графики подобно тому, как это было сделано с «дальней дорогой».

В отличие от полноценной реализации смены текстур дороги, текстура земли (воды) меняется несколько упрощенно, по целым строкам экрана (8 линий), то есть совсем не согласуется с перемещением сегментов дороги. В динамике, опять же, смотрится нормально. Как и многие другие недоработки, это также было улучшено во второй части ТТТ.

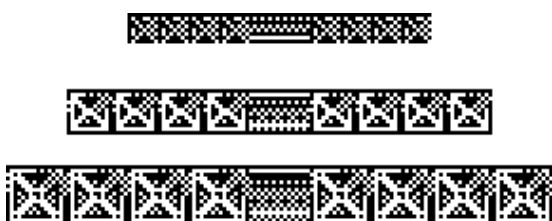
### 140.

На подъемах можно заметить, что линия горизонта не совсем плоская, она поднимается к краям экрана. Это ещё один дополнительный графический элемент (подобный дорисованной воде и «дальней дороге»), который улучшает визуал. Он включается только при необходимости и хорошо сочетается и с задним планом, и с панорамами.

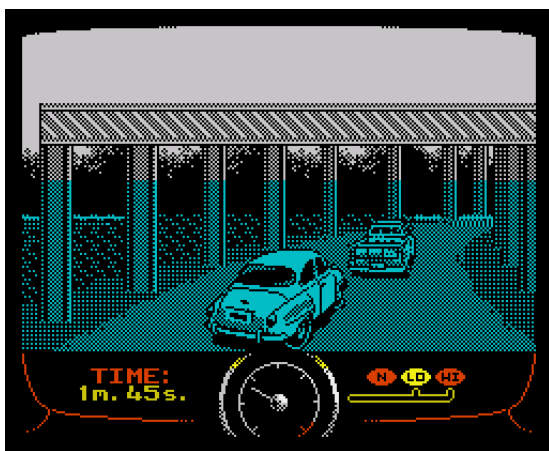


## 141.

Первый вариант конструкций мостов был более детальным:



В игре используется упрощенная версия, в которой горизонтальные конструкции процедурные:



Координаты горизонтальных перекладин мостов привязаны к вершинам стоек. Поскольку ближние перекладины на экране перекрывают все последующие, то только они и выводятся. Понадобилось также учесть тот случай, когда стойки с одной стороны дороги не попадают на экран: ведь именно экранные координаты используются для позиционирования перекладин. В этом случае можно ориентироваться по ближней попавшей на экран стойке, и рисовать перекладину в противоположную сторону до конца экрана.

## 142.

Хотя игра изобилует кат-сценами, но локаций, в которых разворачивается действие, очень немного. Есть те, что встречаются единожды, но в основном они используются для нескольких сцен.



Эта лесная локация—одна из наиболее часто повторяющихся. Здесь развиваются отношения персонажей в течение значительной части сюжета. Две другие часто встречающиеся локации—дом Свена снаружи (начиная со вступительной сцены), а затем и внутри.

В начале работы над игрой мы не знали, как будут выглядеть кат-сцены. Эта находка—сочетание крупных портретов персонажей с весьма мелкой графикой, показывающей действия,—позволила нам уместить в памяти так много сцен.

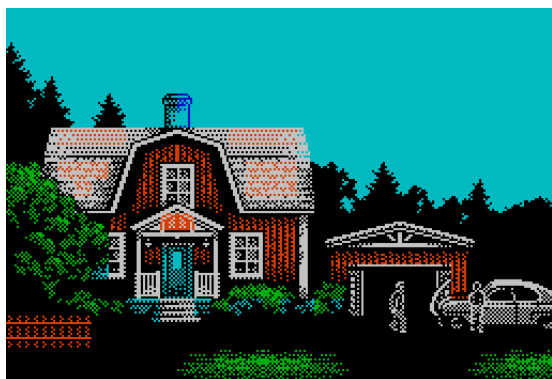
## 143.

Вы также наверняка уже обратили внимание, что кат-сцены используют в том числе и геймплейную графику. В лесной локации тот же задний план, что используется в гонках, те же машины соперников, а спрайт травы используется и в локации с домом Свена. Его же машина в виде сбоку нарисована специально для кат-сцен (в спрайте изображена только видимая часть, и есть еще кусок):



В сцене ремонта машины открытый капот нарисован поверх закрытого, часть

которого закрашена. На какие только ухищрения не приходилось идти ради оптимизации! Дополнительный кусок машины тоже нужен здесь, лишь для её удачного позиционирования у гаража.



144.

Сцена на берегу океана почти целиком состоит из графики геймплея. Добавлены только фигуры Мари и Свена и небольшой кусок скал.



А в этой кат-сцене с лесовозом используется не только полностью геймплейная графика, но еще и часть процедур движка.



145.

В кат-сценах в доме Свена при наличии нескольких персонажей трудно было бы определить, кому принадлежит реплика. Добавление цветной полосы под персонажем решило эту задачу.



146.

Кстати, а вы обратили внимание на цветовое кодирование диалогов? Свен всегда «говорит» жёлтым цветом, дядя Бьёрн—зелёным, Мари—голубым, Ирма и Эва—пурпурным (для Ирмы есть нелепая ошибка в сцене с лесовозами, которую вы только что видели), а Йоханнес—белым.

147.

Готовились спрайты для жёсткой аварии с разворотом разбитой машины, но от этой идеи отказались.



148.

В одной из последних сцен Свен и Мари, сидя за столом, читают газету. Поначалу они были, как и прежде, изображены по разные стороны стола, а в руки Свену пытались поместить газету. Получилось как-то некрасиво, и неожиданно пришла мысль вообще не рисовать никакую газету (она как бы лежит на столе), а вместо этого пересадить Мари к Свену.



И ведь сработало! Они как будто смотрят в одну точку. Обошлись минимумом дополнительных кусков спрайтов.

## 149.

Картинка с Эвой, сидящей на подоконнике, была перерисована специально для этого сделанной фотографии.



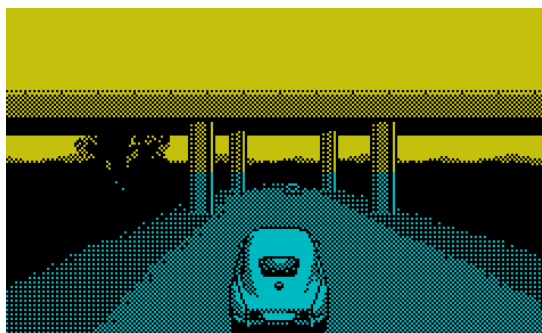
Для оптимизации вертикальные части окон сделаны повторяющимся спрайтом:



## 150.

Эстакады, под которыми проезжают автомобили—еще один очень интересный объект. По реализации они чем-то похожи на мосты, а чем-то на туннели.

Над двумя парами опор с помощью процедур рисуются перекрытия, масштаб и перспектива которых соответствуют их расположению.



При проезде под ними они закрывают небо, но самая интересная деталь—это то, что они отбрасывают тень на машины, причем она появляется и уходит динамически.



Это реализовано путем наложения на уже полностью готовую картинку процедурной текстуры, которая совпадает с текстурой дороги. Она накладывается на тот участок, где нарисован автомобиль, и, разумеется, подействует не только на него, но и на все объекты, которые под эту текстуру попадут. Автомобиль «потемнеет», а вот дорога не поменяется.

Кроме того, если сбоку от машины игрока есть другие автомобили, проезжающие под эстакадой одновременно с игроком, то рисуются дополнительные куски такой текстуры, которые призваны закрыть эти машины.



## 151.

Если в начале игры можно бесконечное количество раз попытаться пройти уровень, то на старших уровнях нам дается ограниченное количество попыток до конца игры. Для их визуализации было нарисовано два варианта иконки. В игре используется второй вариант. Процедура отображения оставшихся попыток переключалась из игры «Bonnie and Clyde» вместе с названием файла. Картинка с машинкой и стрелочкой называется “bc\_hat”, так как в игре про котиков жизни отображались в виде шляп.



## 152.

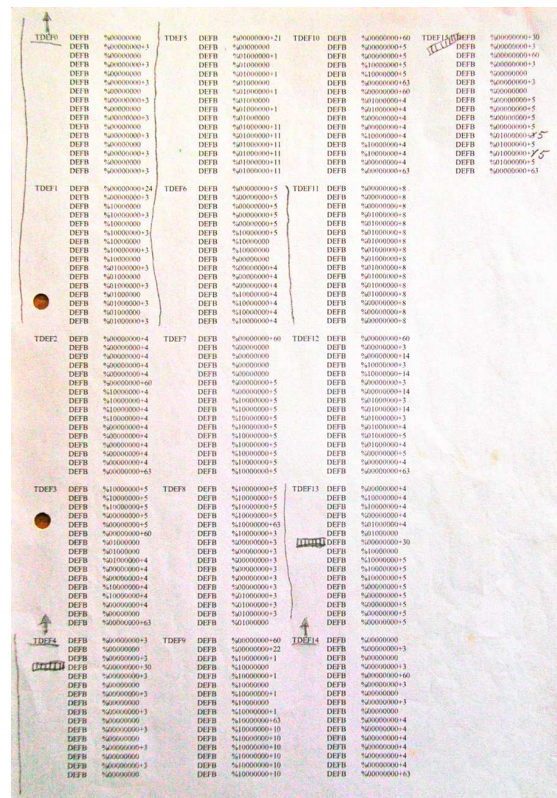
Если в соревновании на время мы пересекли-таки финишную черту или отсечку сектора, но уже после истечения времени, нам засчитают поражение. Что логично, хотя во многих играх это не так.

В дуэли при визуальном одновременном с соперником пересечении финишной черты (и даже если он, вроде бы, чуть впереди) победа, как правило, будет за игроком.

## 153.

Карты уровней всех гоночных игр для восьмибитных компьютеров в принципе довольно однообразны: состоят из похожих прямых участков и поворотов, к которым добавляются все реализованные в игре объекты (если они вообще есть). А ведь однообразие—хороший задел для оптимизации. Чтобы уместить большое количество треков в памяти, мы применяем методы как можно более компактного хранения карт. Во всех играх на движке TRAVEL используется зарезервированная область памяти для текущего трека, в которой перед началом этапа строится карта с помощью процедур и достаточно компактных исходных

данных. В TTT1 для этого используется большое количество предопределенных последовательностей сегментов. На следующем листе представлена «типовая» цепочка из 256 сегментов, поделенная на 16 участков. На этих сегментах есть повороты и прямые участки, указаны придорожные объекты, некоторые знаки, пешеходные переходы и подъемы. В карте игры можно просто указать: «использовать типовую последовательность, начиная с участка #0 и продолжительностью 3 участка».



Причём, при других изменениях на трассе (скажем, въезде в туннель или начале узкой дороги) могут использоваться те же данные, но выглядеть на экране это будет несколько иначе: например, в туннелях, останутся только повороты.

Мы не постеснялись использовать этот же массив почти без изменений в играх Rubinho Cucaracha и Travel Unlimited.

Но это не единственный вариант заранее заданных последовательностей сегментов в карте. Есть еще перечень коротких, но более уникальных цепочек. Вы видели его на одном листе со списком всех объектов. Так, например, задаются эстакады с правильным подъездом к ним, короткие

туннели, разные виды мостов, длинные ряды столбов и так далее.

А ещё есть возможность обращения к карте другого трека. Скажем, памяти остается всё меньше, а что там было в 5-м уровне, игрок уже вряд ли помнит, находясь на 55-м. Тем более что и тип соревнования другой, и машины, и задний план, и цветовые атрибуты, а может, это ночь или зима... Можно просто написать команду перехода к данным по указанной метке и повторить часть другого трека.

#### 154.

Анимация персонажей для кат-сцен собрана в одном файле.



Кстати, для ходьбы вверх и вниз персонажи используют всего по одному спрайту, который отзеркаливается.

#### 155.

Для выезда машины из гаража используется набор спрайтов с меняющимся освещением машины.



Подобно этому, Свен, заходя в гараж, исчезает в тени.



#### 156.

Очевидно, что визуализация столкновений с отлетающими обломками сделана по примеру игры Chase H.Q. Обломки

используют анимацию из четырех кадров и отлетают по различным вариантам траектории.



#### 157.

Ночные гонки геймплейно и технически ничем не отличаются от дневных. Для неба лишь используется более темный цвет, но главное визуальное отличие—конечно, свет фар.



Это реализовано с помощью одного лишь особым образом нарисованного спрайта: само изображение в нем «белое», а маска клетчатая.



При выводе этого спрайта включенные «белые» пиксели света фар всегда попадают на «черные» пиксели дороги, и на дороге образуется сплошное светлое пятно. В то же время, другие объекты будто бы действительно «освещаются», но не зарисовываются полностью, ведь на них тоже попадает сеточка «белых» пикселей.



И лишь спустя время после выхода игры, при работе над второй частью, мы поняли, что вообще-то сильно перемудрили с текстурой «света», которая должна быть намного проще:



В этом случае на машинах в свете фар уже не будет появляться ошибочный сетчатый прямоугольник, который замечен в ТТТ1.

Во второй части игры используется уже исправленная текстура.

## 158.

Для трактора дяди Бьёрна, как и для лесовозов, используются собственные спрайты для дальних дистанций, потому что количество прорисованных масштабов больше, чем для автомобилей.



А для двух ближних расстояний прорисовано вращение колес: каждый второй кадр поверх уже нарисованного спрайта выводятся дополнительные куски.

На ближнем расстоянии часть торса тракториста для оптимизации нарисована отдельным спрайтом гораздо меньшей ширины.

## 159.

Для вывода обратного отсчета мы использовали шрифт с динамическим эффектом, что относительно редко встречается в играх для ZX Spectrum. Появляющиеся на экране цифры увеличиваются в размерах. Для этого мы использовали два набора символов и возможности одного из наших спрайтовых форматов: сначала на очень короткое время выводятся цифры меньшего размера, а затем—более крупные, причем для них действует постепенное растяжение по вертикали (используется тот же самый формат спрайта, что и для машин соперников). А для надписи «GO!» мы ограничились одним размером шрифта.

1 2 3 4 5 6 7 8 9 GO!  
1 2 3 4 5 6 7 8 9

## 160.

Работы над заставкой и постером велись одновременно. Конечно, использовались уже имеющиеся 3D модели автомобилей. Далее, на основе фотографий (вообще-то это Норвегия, а не Швеция, но часть сюжета игры проходит и там)...



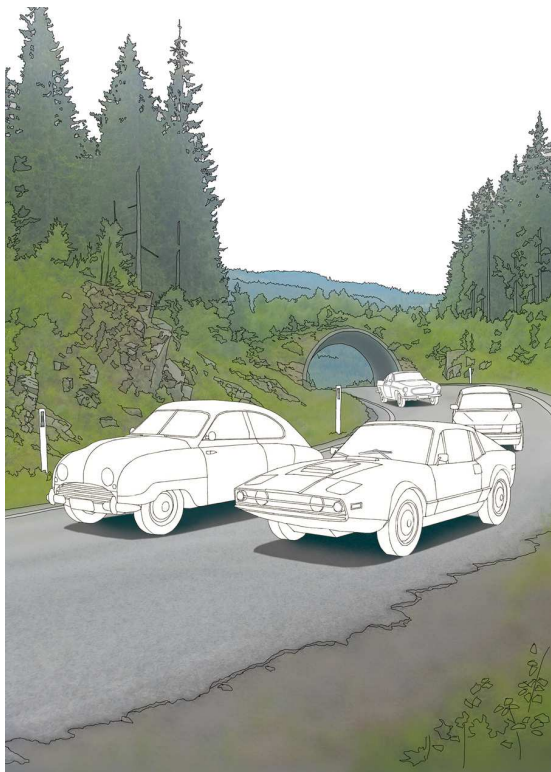
...подготовили исходное изображение заднего плана постера...



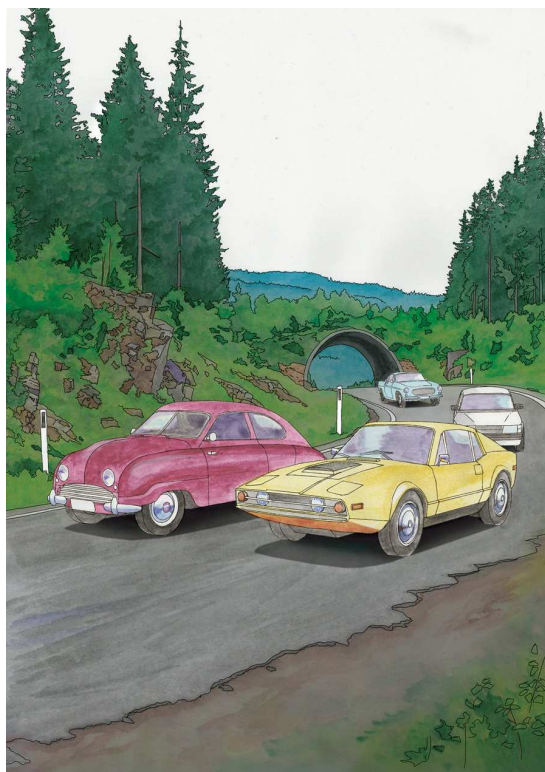
...нарисовали на бумаге акварельный набросок...



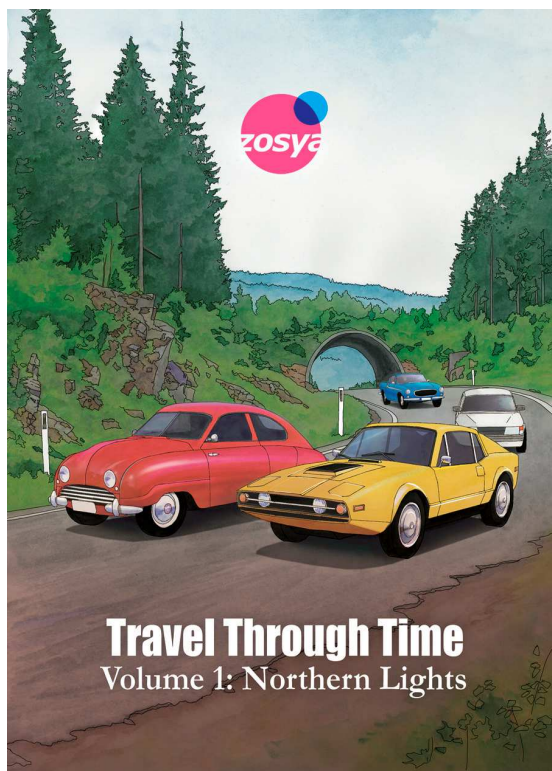
...сделали контурные обводки, размытие, смонтировали рендеры 3D моделей...



...объединили с уже имеющимся изображением...

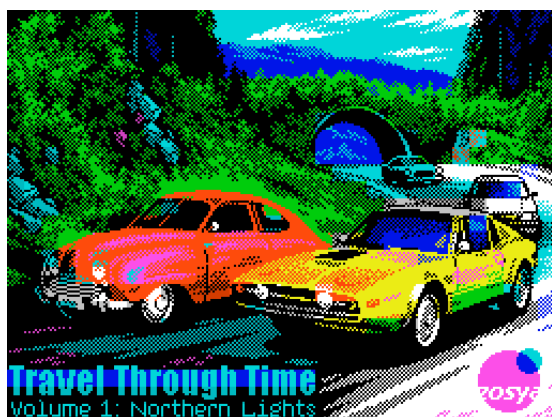


...сделали множество цифровых дорисовок и получили финальный постер:



161.

И подобные же методы для создания заставки на ZX Spectrum:



162.

Проект финальной сцены содержал изображения деревьев. Не уместились в памяти. Да и не слишком украшали картинку. Можно также увидеть более детальную анимацию отражений. В игре вместо этого используется один отзеркаливающийся спрайт.



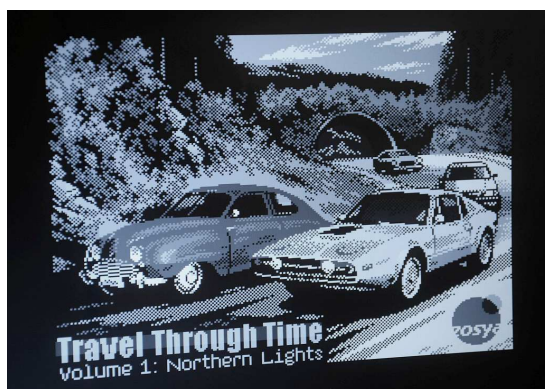
163.

Трейлер игры—самый популярный из всех наших видеороликов.



164.

Одна из наших любимых «забав»: проверять, как смотрится графика игры (и особенно заставка) в монохромном варианте. Ведь именно так многие из нас видели компьютерные игры в конце 80-х—начале 90-х.



## 165.

Довольны ли мы результатом? Далеко не о каждой игре мы можем так сказать, но в данном случае—однозначно да. Travel Through Time Volume 1: Northern Lights мы определенно можем считать (одной из) лучших наших игр и (одной из) лучших гоночных игр для ZX Spectrum. Продажи показали, что сообщество с нами солидарно.

