

# Ядро системы генерации многослойных динамических титров

И. М. Арсенин, И.Г. Таранцев  
Институт Автоматики и Электрометрии СО РАН,  
Новосибирск, Россия  
egor@sl.iae.nsk.su

## Аннотация

Рассмотрены особенности реализации системы генерации многослойных динамических титров в реальном масштабе времени на ПК под управлением ОС Windows. Рекомендуется алгоритм микширования многих слоев. Показана возможность реализации универсального ядра, не зависящего от конкретного устройства отображения титров.

**Ключевые слова:** мультимедиа, многослойные титры, реальное время.

## 1. ВВЕДЕНИЕ

Производительности современных персональных компьютеров достаточно для обработки в реальном масштабе времени одного или нескольких потоков видео и звука. Существует большое количество разнообразных устройств ввода-вывода мультимедиа потоков. С другой стороны существует потребность в простых, удобных системах наложения текстовой и графической информации на видеоизображение. Причем в последнее время одним из основных требований к системам генерации титров становится требование постоянного расширения возможностей при неизменном аппаратном обеспечении. Поэтому все чаще и чаще используются системы генерации титров на базе персонального компьютера. Результатом работы любой системы генерации титров является двумерное изображение с прозрачностью, накладываемое на внешнее видеоизображение. Как правило, это изображение формируется из нескольких независимых или связанных общим поведением слоев. Существуют системы, формирующие изображение из трехмерной сцены. Однако такие системы зачастую не используются пользователями из-за сложности управления трехмерными объектами. Например, при формировании «бегущей строки», курса валют и других простых титров достаточно указать текст и время показа. Богатые возможности предполагают относительно сложный пользовательский интерфейс. Но даже в этом случае можно считать, что каждый слой, создаваемый 3D-системой представляет собой плоское динамическое изображение. Таким образом, для любой многослойной титровой системы основной задачей ее ядра можно считать обеспечение эффективного микширования произвольного количества слоев, каждый из которых представляет собой последовательность двумерных изображений с прозрачностью.

К сожалению, в ОС Windows нет стандарта для наложения текстовой и графической информации (наложения титров). Попытки использовать механизмы Direct Show к успеху не привели. Основными недостатками Direct Show являются:

- Большое количество потоков управления, порождаемых объектами. Это приводит к большим накладным расходам, связанным с переключением между активными потоками управления. При стандартной частоте переключения около 20 Гц очень трудно получить время реакции системы на уровне 25 Гц.
- Высокие требования к ресурсам компьютера, предъявляемые объектами, входящими в стандартную поставку ОС Windows. В первую очередь это касается объема оперативной памяти, необходимой для нормальной работы стандартных объектов.
- Невозможность динамического переключения потоков мультимедиа-данных между объектами, а также невозможность добавления или удаления объектов в процессе работы. Это приводит к тому, что все возможные типы объектов необходимо активировать сразу в начале работы, что очень сильно повышает требования к ресурсам компьютера. Это также заставляет разработывать свои собственные объекты «маршрутизаторы», что, естественно, занимает дополнительное время.
- Невозможность управления временем работы того или иного объекта Direct Show. Это приводит к тому, что неиспользуемый сейчас объект отнимает ресурсы у объекта, выполняющего реальную работу.

Поэтому нами было принято решение разработать собственное ядро системы генерации титров, обеспечивающее микширование в реальном времени произвольного количества слоев, каждый из которых представляет собой последовательность двумерных изображений.

## 2. РЕАЛИЗАЦИЯ ЯДРА

### 2.1 Обеспечение генерации титров в реальном масштабе времени

Поскольку ОС Windows не является системой реального времени при работе с частотами 25-30 Гц, то для непрерывности воспроизведения титров (последовательности изображений) необходимо использовать дополнительный кадровый буфер (ФИФО), сглаживающий задержки в процессе генерации последовательности изображений конкретным генератором титров. Глубина буфера должна определяться характеристиками конкретного ПК, а также задачами и ограничениями, задаваемыми пользователем. В большинстве случаев достаточно буферизации 10-15 кадров.

## 2.2 Алгоритм микширования

Рассмотрим стопку из  $n$  слайдов, каждый из которых имеет цвет  $C_n$  и непрозрачность  $A_n$ . С учетом только верхнего слайда результирующий цвет стопки равен  $C_{res} = C_1 \cdot A_1 + C_{back} \cdot (1 - A_1)$ , где  $C_{back}$  - цвет фона. С учетом второго слайда

$$C_{res} = C_1 \cdot A_1 + C_2 \cdot A_2 \cdot (1 - A_1) + C_{back} \cdot (1 - A_2) \cdot (1 - A_1)$$

Аналогично с учетом всех  $n$  слайдов результирующий цвет стопки вычисляется по следующей формуле:

$$C_{res} = C_1 \cdot A_1 + C_2 \cdot A_2 \cdot (1 - A_1) + \dots + C_n \cdot A_n \cdot \prod_{k=1}^{n-1} (1 - A_k) + C_{back} \cdot \prod_{k=1}^n (1 - A_k) \quad (1)$$

Если заменить стопку слайдов одним цветом  $C_x$  с непрозрачностью  $A_x$ , то для него можно написать:

$$C_{res} = C_x \cdot A_x + C_{back} \cdot (1 - A_x) \quad (2)$$

Из сравнения (1) и (2) можно найти непрозрачность и цвет стопки слайдов:

$$A_x = 1 - \prod_{k=1}^n (1 - A_k) \quad (3)$$

$$C_x = \frac{C_1 \cdot A_1 + C_2 \cdot A_2 \cdot (1 - A_1) + \dots + C_n \cdot A_n \cdot \prod_{k=1}^{n-1} (1 - A_k)}{1 - \prod_{k=1}^n (1 - A_k)} \quad (4)$$

Рассмотрим пару величин: цвет, умноженный на непрозрачность (premultiplied color)  $P = C \cdot A$  и величину, обратную прозрачности, т.е. прозрачность  $Tr = 1 - A$ . Для этой пары величин формулы (3) и (4) приобретут следующий вид:

$$Tr_x = \prod_{k=1}^n Tr_k \quad (5)$$

$$P_x = P_1 + P_2 \cdot Tr_1 + \dots + P_n \cdot \prod_{k=1}^{n-1} Tr_k \quad (6)$$

Видно, что пара формул (5) и (6) может быть легко реализована итеративно при последовательном микшировании произвольного числа слайдов, начиная с самого верхнего.

## 2.3 Микширование многих слоев

При одновременном отображении нескольких изображений, возможно перекрывающихся на экране, необходимо однозначно определить порядок их отображения. Традиционно для этого вводят целочисленную координату  $Z$ , либо понятие слоя со своим номером. Чем больше номер слоя (координата  $Z$ ), тем «выше» расположен объект (вышележащий объект перекрывает все нижележащие объекты).

Для однозначного смешивания изображений разных генераторов титров мы ввели ограничение – в каждом слое может располагаться только одно изображение. Это ограничивает число возможных слоев примерно 4 миллиардами, что не является реальным ограничением на

практике, поскольку реальная площадь титров обычно не превышает однократной площади экрана (повторная заливка приводит к частичному затиранию нижележащих титров).

Стандартным форматом для представления цвета с прозрачностью (альфа-каналом) является формат RGBA, поэтому именно этот формат и был выбран в качестве формата данных для хранения изображений в ФИФО.

В итоге был выработан интерфейс ITitleRegion на базе стандарта COM, обеспечивающий унифицированный доступ к ФИФО кадров. Кроме того, в интерфейсе предусмотрена возможность дополнительно назначать общую непрозрачность изображений в данном ФИФО. Это позволяет генератору титров плавно проявить или спрятать титр, не изменяя изображения в ФИФО и, следовательно, не передавая новых данных между генератором титров и ядром титровальной системы, экономя тем самым ресурсы компьютера.

## 3. РЕАЛИЗАЦИЯ УНИВЕРСАЛЬНОГО ЯДРА

Рассмотрим работу ядра по микшированию очередного кадра. Для формирования результирующего изображения необходимо опросить состояние всех регионов (ФИФО кадров) и определить текущее положение прямоугольника с изображением для каждого ФИФО. Затем необходимо разбить площадь экрана на непересекающиеся прямоугольники, в каждом из которых будет располагаться либо один слой, либо несколько слоев, либо ни одного слоя. Соответственно, в первой случае надо просто скопировать часть изображения из ФИФО; во втором случае надо смикшировать все изображения вышеописанным алгоритмом; в последнем случае просто обнулить часть кадрового буфера. Далее сформированный кадровый буфер может быть передан для отображения. Заметим, что только последний этап предполагает общение с конкретным устройством отображения видеoinформации. Наиболее сложные этапы микширования изображения и отслеживания состояний всех ФИФО одинаковы для любого устройства. Это позволило выделить в отдельный интерфейс все методы общения с конкретным устройством отображения видеoinформации ITitleSurface. В настоящий момент реализовано несколько вариантов титровальных поверхностей (объектов ITitleSurface): на базе обычного окна Windows, на плате ФорвардТ и в качестве фильтра Direct Show.

В итоге, работа системы автоматизации (или любого другого приложения), использующей титры состоит из следующих этапов:

- Определение доступных титровальных поверхностей (объектов ITitleSurface) и выбор одного из них.
- Создание титрованного ядра на базе выбранного объекта.
- Динамическое создание генераторов титров, каждый из которых создает в титровальном ядре свое ФИФО кадров и управляет им.

При этом возможно создание, удаление и изменение любых параметров титровальных регионов (ФИФО кадров) непосредственно в процессе работы титровального ядра. Каждый из регионов управляется независимо.

#### 4. ОПТИМИЗАЦИЯ

При работе титровального ядра больше всего ресурсов компьютера занимает процесс формирования очередного кадра. При этом используются алгоритмы копирования и микширования нескольких изображений. Нами был написан ряд процедур, использующих возможности дополнительного набора команд процессоров Intel и AMD, а именно, наборов команд MMX, SSE и SSE2. Это позволило значительно повысить скорость формирования очередного кадра.

Также, в процессе анализа состояний ФИФО кадров проверяется обновление ФИФО. Любая часть экранного буфера микшируется (копируется) заново только при изменении одного из составляющих ее изображений.

В результате система обеспечивает микширование практически неограниченного количества слоев общей площадью до двух экранов и при этом занимает не более 20% ресурсов компьютера на базе процессора Pentium 4 3 GHz (HT). Т.е. у системы остается достаточно ресурсов для генерации сложных динамических титров и для воспроизведения полноэкранного изображения, на фоне которого будут отображаться генерируемые титры, а также для надежной работы системы автоматизации, управляющей процессом формирования конечного изображения.

#### Об авторах

Игорь Геннадьевич Таранцев – научный сотрудник лаборатории 13 Института Автоматики и Электростроения СО РАН.

Адрес: Новосибирск, 630090, пр-т Лаврентьева, 1а.

Телефон: (383) 33-99-220

E-mail: [egor@sl.iae.nsk.su](mailto:egor@sl.iae.nsk.su)

Арсенин Игорь Михайлович – ведущий инженер-программист лаборатории 13 Института Автоматики и Электростроения СО РАН.

Адрес: Новосибирск, 630090, пр-т Лаврентьева, 1а.

Телефон: (383) 33-99-220

E-mail: [ima@sl.iae.nsk.su](mailto:ima@sl.iae.nsk.su)

## Core of multilayered dynamic title generation system

### Abstract

Describe realization features of multilayered dynamic title generation system for OS Windows. Show method for real time guaranty. Recommend mixing algorithm. Separate core, which is not dependent on the concrete titling device.

**Keywords:** *Multimedia, multilayered title, real time.*

### About the authors

Igor Tarantsev is research officer of laboratory 13 at Institute of Automation and Electrometry. His contact email is [egor@sl.iae.nsk.su](mailto:egor@sl.iae.nsk.su).

Igor Arsenin is leading engineer of laboratory 13 at Institute of Automation and Electrometry. His contact email is [ima@sl.iae.nsk.su](mailto:ima@sl.iae.nsk.su).