

Достоинства и недостатки использования ассемблерных вставок для современной игровой индустрии

Перов Даниил Андреевич, Абрамова Оксана Фёдоровна.

Волжский политехнический институт (филиал) ФГБОУ ВПО "Волгоградский государственный технический университет" Волжский, Россия

Аннотация

В данной статье будет рассмотрен язык ассемблера с точки зрения полезности для современной игровой индустрии.

Ключевые слова: Ассемблер, игры, графика, обработка.

Заголовок на английском

Perov Daniel Andreivich, Abramova Oksana Fyodorovna

Volzhskiy Polytechnical Institute, branch of the Volgograd State Technical University Volzhskiy, Russia

Ключевые слова на английском: Assembler, games, graphics, handling.

Введение

Язык Assembler стоял у истоков современной игровой индустрии. Стоит напомнить, что именно на ассемблере были написаны игры для консолей Dendy и Sega. Это была эпоха платформеров с пиксельной графикой. Ограниченность игровых систем вынуждала разработчиков экспериментировать с моделями геймплея. Так мы получили достаточно много игровых жанров, которые применимы по сей день. Но со временем игрокам надоела однообразная пиксельная картинка и разработчики, один за другим, пытались удивить графическими новшествами.

Недостатки ассемблера

Если нужно выбрать между ассемблером и двоичным кодом, то для программиста ассемблер смотрится выгоднее и проще, но приближённый к машинным кодам язык, имеет свои минусы.

Разработчику игр приходится постоянно следить за вызовами определённого участка кода, т.к. ассемблер не имеет никаких фигурных скобок для определения границ выполнения функции.

Код на ассемблере выглядит не самым понятным образом. Здесь спасением могут стать комментарии к коду.

Пример ассемблерного кода

```
mov ah,0
mov al,x
cmp al,7
jg MoreThanSev
cmp al,-7
jle LessThanMSev
mov ah,9h
lea dx,messops
int 21h
jmp exit
```

Всё дальше от ассемблера

Технологии развивались, как и языки программирования. Больше не нужно было чередовать пиксели для экономии цветовой палитры. Появились графические интерфейсы и API для упрощения работы разработчиков. Все эти нововведения помогли не только повысить качество игрового процесса, но и перенести его в трёхмерное пространство.

Пример кода OpenGL

```
void display()
{
    glClear(GL_COLOR_BUFFER_BIT);
    glBegin(GL_QUADS);
    glColor3f(1.0, 1.0, 1.0);
    glVertex2i(250, 450);
    glColor3f(0.0, 0.0, 1.0);
    glVertex2i(250, 150);
    glColor3f(0.0, 1.0, 0.0);
    glVertex2i(550, 150);
    glColor3f(1.0, 0.0, 0.0);
```

```
glVertex2i(550, 450);  
glEnd();  
glutSwapBuffers();  
}
```

Ещё позднее игры приблизились к уровню кино, многолетняя погоня за качеством графики дала свои плоды. И дело не только в графике, арсенал разработчиков значительно пополнился. Экспозиция, эффекты, освещение, постановка кадра, всё это придало картинке на экране эффект погружения.

Сила ассемблера

С развитием игровой индустрии ассемблер никуда не исчез, он занял своё законное место как посредник между графическим процессором и интерфейсом, последний разработчик использует для применения графических эффектов. Если посчитать, что каждый вызов графического интерфейса подразумевает инструкцию для GPU, то без ассемблера о графике как таковой можно забыть.



- 1 - чип проверки региона
- 2 - банк памяти с графическими данными
- 3 - банк памяти с кодом игры

В картриджах для игровых консолей была одна любопытная особенность. Сама игра была поделена на графические данные и код самой игры. Теперь блок графических данных заменили интерфейсы, а транслированием их кода занимается ассемблер.

Заключение

Игровая индустрия прошла долгий путь от пиксельного арта до современной графики. Весь этот путь начался и продолжается до сих пор благодаря ассемблеру.

Библиографический список

1. **Абрамова О.Ф.** Компьютерная графика. Избранные лекционные темы. Ч. 1: учеб. пособие / О.Ф. Абрамова. - Saarbrücken, 2012. - 129 с.
2. Создание графических примитивов на Ассемблере [Электронный ресурс] // URL: <https://habr.com/post/134495/>
3. Графика на ассемблере [Электронный ресурс] // URL: <http://www.chipinfo.ru/literature/chipnews/200007/48.html>
4. **Яблоков К. М.** Исторические компьютерные игры как способ моделирования исторической информации // История и математика: Анализ и моделирование социально-исторических процессов. М.: КомКнига, 2007. С.170-204.