

## **АПАРАТНА ТЕСЕЛЯЦІЯ В DIRECTX 11 ТА OPENGL 4**

*Розглянуто організацію апаратної теселяції відеокарт з використанням DirectX 11 та OpenGL 4.0.*

### **Вступ**

Формування реалістичних тривимірних моделей потребує збільшення кількості полігонів у моделях та, як наслідок, підвищення вимог до швидкості передачі інформації у графічний адаптер. Використання теселяції дозволяє надсилати у графічний адаптер модель з відносно малою кількістю полігонів і формувати високополігональну модель безпосередньо під час рендерінгу. З появою графічних інтерфейсів DirectX 11 і OpenGL 4.0 і відповідних змін у архітектурі графічних адаптерів стало можливим перенесення обчислення алгоритмів теселяції з центрального (CPU) на графічний (GPU) процесор. Відеокарти корпорації AMD підтримують DirectX 11 і OpenGL 4.0 починаючи з моделей серії Radeon 5000. У графічних адаптерах серії GeForce корпорації NVIDIA підтримку DirectX 11 та OpenGL 4.0 забезпечують моделі, починаючи з GTX480/470 [1, 2].

### **Теселяція в DirectX 11**

Безпосередній процес теселяції в Direct3D 11 виконується за допомогою трьох нових стадій графічного конвеєра: програмованих Hull Shader та Domain Shader і конфігурованої Tessellator. Схематично новий графічний конвеєр DirectX 11 наведено на рис. 1.

Hull Shader отримує дані про контрольні точки кожного патча з вершинного шейдера (Vertex Shader), при необхідності модифікує їх, та одночасно обчислює ступінь теселяції, який визначає рівень розбиття кожного патча. Після цього інформація про контрольні точки передається в Domain Shader, а коефіцієнти теселяції - в конфігурований теселятор (Tessellator). Програмованість Hull Shader дозволяє проводити адаптивне визначення коефіцієнтів теселяції, за рахунок чого поверхня тривимірного об'єкта може мати різну деталізацію залежно від положення в сцені [3, 4].

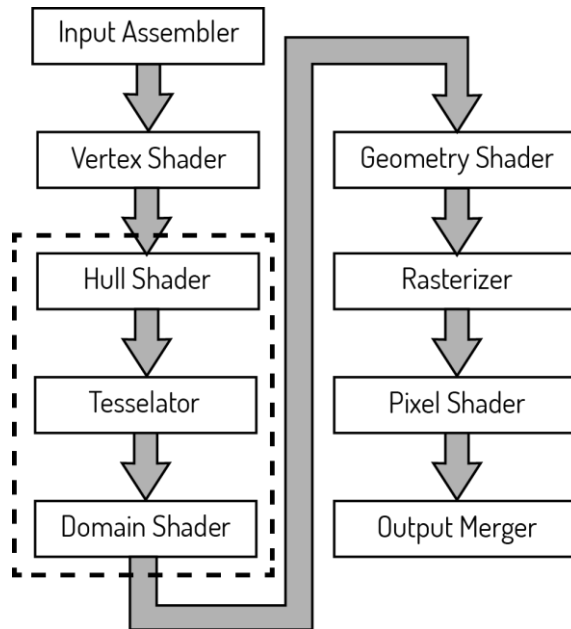


Рисунок 1 – Графічний конвеєр DirectX 11.

Головна функція теселятора полягає у розділенні патча на декілька примітивів трикутної чи квадратної форми базуючись на інформації, отриманій від Hull шейдера. Після завершення роботи теселятор передає інформацію про нову топологію патча Domain шейдеру.

Domain шейдер використовує дані про контрольні точки патча, а також коефіцієнти теселяції від Hull шейдера, та інформацію про нову топологію патча, отриману від теселятора, для формування вершин, які далі передаються в геометричний шейдер (Geometry Shader).

Написання коду для шейдерів Hull та Domain виконується за допомогою мови HLSL.

### Теселяція в OpenGL 4.0

Шейдери теселяції та геометрії в OpenGL 4.0 дають програмістам додаткові можливості модифікації геометрії тривимірної моделі по мірі її проходження через графічний конвеєр. На рис. 2 зображено графічний конвеєр OpenGL 4.0 [5].

Безпосередню теселяцію в наведеному вище конвеєрі виконують шейдери Tessellation Control і Tessellation Evaluation та Tessellation Primitive Generator.

Основне призначення шейдера Tessellation Control полягає у налаштуванні теселятора (визначення кількості примітивів, які будуть згенеровані теселятором, та алгоритму, за яким буде проводитись їх створення) і формування вихідних атрибутів кожної вершини.

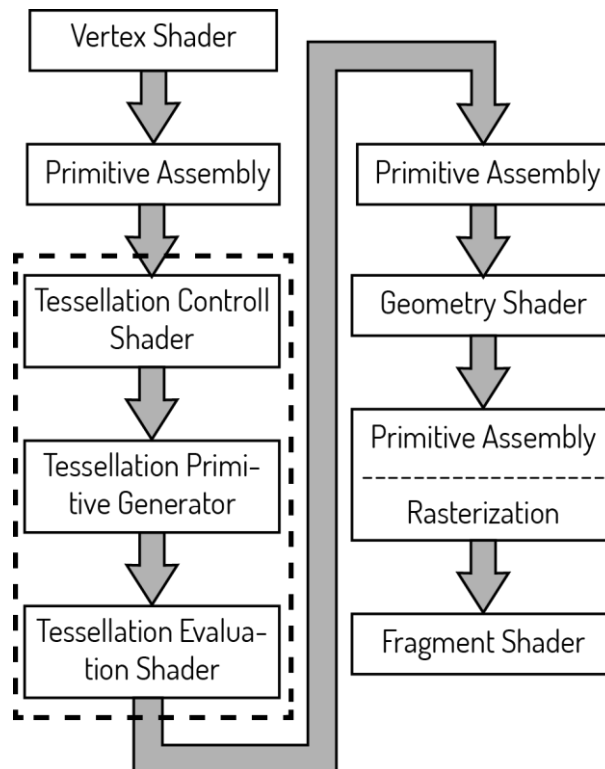


Рисунок 2 – Графічний конвеєр OpenGL 4.0.

Генератор примітивів (TPG), як і в DirectX 11, є конфігурованою стадією конвеєра, яка створює примітиви для подальшого використання в конвеєрі за допомогою набору стандартних алгоритмів теселяції.

Для визначення позиції та іншої необхідної інформації (вектору нормалі, координати текстури та ін.) кожної вершини примітиву, сформованого теселятором, застосовується шейдер Tessellation Evaluation.

Написання коду для шейдерів Tessellation Control і Tessellation Evaluation виконується за допомогою мови OpenGL Shading Language - GLSL.

### Висновки

Використання теселяції підвищує реалістичність об'єктів у тривимірній сцені. Поява підтримки апаратної теселяції сучасними відеокартами і найпопулярнішими графічними інтерфейсами DirectX і OpenGL дозволяє створювати програмне забезпечення, яке проводить рендер складних моделей в реальному часі за допомогою теселяції.

### Література

1. ATI Radeon™ HD 5870 Graphics [Електронний ресурс] – Режим доступу: <http://www.amd.com/us/products/desktop/graphics/ati-radeon-hd-5000/hd-5870/Pages/ati-radeon-hd-5870-overview.aspx>
2. GeForce GTX480 Built for DirectX11 Tessellation [Електронний ресурс] – Режим доступу: <http://www.geforce.co.uk/hardware/desktop-gpus/geforce-gtx-480/features>

3. Tessellation Overview [Электронный ресурс] – Режим доступа: <http://msdn.microsoft.com/en-us/library/windows/desktop/ff476340%28v=vs.85%29.aspx>
4. Direct3D 11 Details Part II: Tessellation [Электронный ресурс] – Режим доступа: <http://www.realtimerendering.com/blog/direct3d-11-details-part-ii-tessellation>
5. David Wolff OpenGL 4.0 Shading Language Cookbook / David Wolff. – Packt Publishing, 2011. – 340 с. – ISBN 978-1-849514-76-7