

Spielekonsolenprogrammierung

Text und Dreieck Rendern

- Verwendung von libdbgfnt
- Aufbau eines ersten interaktiven Programmes
- Zeichnen eines Dreiecks
- Verwendung der Mathebibliothek (Manipulation des Dreiecks)

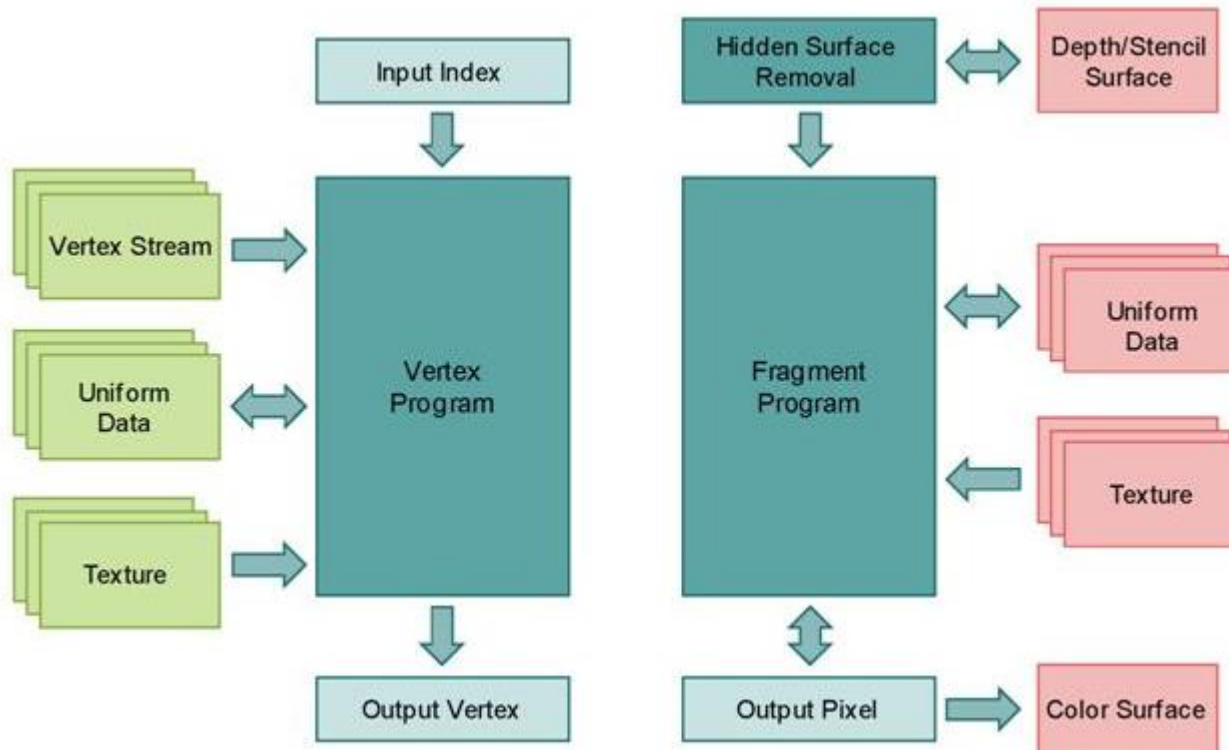
- Wird verwendet als Debug Ausgabe
 - Benutzt die CPU um in den Framebuffer zu schreiben!
 - Konsequenz: Ausgabe erfolgt in der Sync Routine.
 - Sollte nicht in fertigen Programmen verwendet werden
- Benutzte Dateien:
 - Header: libdbgfnt.h
 - Lib: libSceDbgFont.a

- Benutzung
 - Initialisierung: **sceDbgFontInit**
 - Ausgabe: **sceDbgFontPrint**
 - Flush (eigentliches zeichnen): **sceDbgFontFlush**
 - Abschalten (wenn Programmende vorgesehen): **sceDbgFontExit**
- Vorführung des Hello World Programmes!

- Aufgabe 1:
- Erweitern Sie das Hello World Programm so, dass die Messwerte des linken Analogsticks ausgegeben werden!
- Lesen Sie dazu in der Kernschleife den Wert aus und geben ihn auf dem Display aus!

Zeichnen eines Dreiecks

- Zusammenfassung Rendering Context

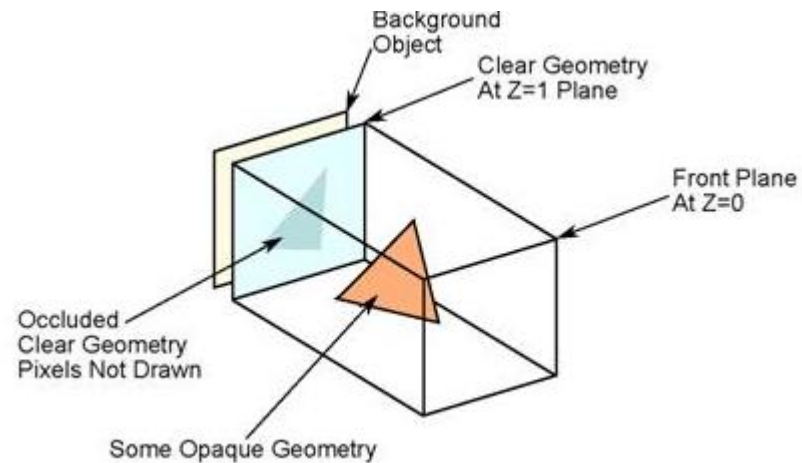


Zeichnen eines Dreiecks

- Clear triangle war schon eines !
- Wir wollen eine Farbe in den Ecken vorgeben!
- Wir wollen das Dreieck später drehen können!
- Benötigt wird:
 - Eine Vertex Definition mit Farbe!
 - Ein separates Fragment und Vertex Programm!
 - Ein separater Rendering Aufruf dafür!

Zeichnen eines Dreiecks

- Was passiert eigentlich in einem Tile Prozessor?



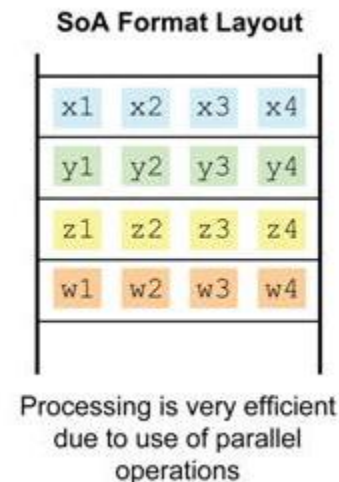
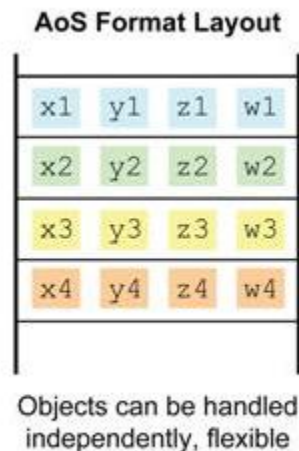
Zeichnen eines Dreiecks

- Vertices:
 - Vertices können alle möglichen Daten enthalten.
 - Hier: Position und Farbe
 - Die Interpretation erfolgt im Shader.
 - Shader werden in der nächsten Stunde näher behandelt.
- Wenn das Dreieck transformiert werden soll, benötigen wir uniform Parameter (= gleich für alle Vertices)

Zeichnen eines Dreiecks

- Schauen wir mal in das Programm!
- Matrizen werde Row Major interpretiert, aber Column Major hochgeladen!
- In dem linearen Array werden nach und nach die einzelnen Zeilen angelegt.
- Aufgabe 2 danach: Machen Sie das Dreieck mit dem linken Joystick drehbar!

- Zu finden unter System->Vectormath Library
- Bildet auf SIMD Einheiten (PC: SSE) ab
- Bietet Vektoren, Matrizen (2x2,3x3, 4x4) und Quaternions an
- Ablage in AoS (array of structures) und SoA (structure of arrays)



- Matrizen Funktionalität:
 - Konstruktion für verschiedene CG Anwendungen (LootAt, Frustrum...)
 - Matrizenmultiplikation
 - Invertierungen
- Quaternion:
 - Lerp
 - Konstruktion
 - Multiplikation etc.

- Aufgabe 3: Führen Sie die für die Drehung des Dreiecks notwendigen Berechnungen mit der Mathebibliothek durch
- Tipp: Wenn Sie AoS verwenden, liegen die Floats in der Matrix schon in der Reihenfolge, die Sie benötigen

- Verwendung von libdbgfnt
- Aufbau eines ersten interaktiven Programmes
- Zeichnen eines Dreiecks
- Verwendung der Mathebibliothek (Manipulation des Dreiecks)