Beginn: 13:00 Uhr
Ort: Arbeitsgruppenraum C2.227
Anwesende: alle Projektgruppenteilnehmer außer Frank Ullmann

Als Tagesordnungspunkte lagen vor:

- Vorstellung des Arbeitsstandes der Gruppe Algorithmusentwurf
- Terminplanung für die Zeit bis zum Vorlesungsende

Zuerst stellte die Gruppe Algorithmusentwurf Ihren bisherigen Arbeitsstand vor. Hierbei wurde vor allem auf die Frage eingegangen, wieweit eine Parallelisierung vorgenommen werden kann. In der Anfangsphase, in der das Gleichungssystem aufgestellt wird, kann parallel gearbeitet werden. Sobald die ersten drei Stützstellen (x- und y-Werte) übergeben wurden, kann die erste Gleichung aufgestellt werden. Nach jeder weiteren übergebenen Stützstelle kann eine weitere Gleichung aufgestellt werden. Durch dieses Übergabekonzept ist es notwendig, daß für jedes Register das valid-Flag einzeln gesetzt wird. Für die Rückgabe werden je Teilpolynom vier Register für die vier Koeffizienten benutzt. Hierdurch ist es sinnvoll, nur ein spezielles valid-Flag für die Rückgabe-Register zu verwenden. Hierbei können auch parallel die Koeffizienten für ein weiteres Teilpolynom berechnet werden, während die Register beschrieben und von der IU ausgelesen werden.

Die Eingabe-Register sollen auch als interne Register für den Rechenkern zur Verfügung stehen. Hierzu ist eine Auswahl mittels Multiplexer notwendig, wonach entschieden wird, ob die Register vom Bus beschrieben werden, oder vom Rechenkern aus. Die Werte aus den Registern können nur vom Rechenkern ausgelesen werden. Für die Ausgabe stehen vier zusätzliche Register bereit.

Bei der Lösung des Gleichungssystems müßen alle Matrixeinträge gleichzeitig zugreifbar sein, da bei den Rechenschritten auf allen Spalten und Zeilen der Matrix Operationen ausgeführt werden. Auch eine Beschränkung auf eine Dreibandmatrix ist nicht möglich, da in Zwischenschritten mehrere Matrixeinträge gemacht werden.

Mögliche Fehler, die bei dem Löesen eines Gleichungssystems auftreten können, müßen vorher von der Benutzerschnittstelle erkannt werden, und von Ihr angefangen werden. In der eigentlichen Berechnung dürfen keine Fehler auftreten können, da keine Behandlung von Fehlern vorgesehen ist.

Zur Realisierung des Algorithmus wurde noch das Verfahren nach dem Horner-Schema vorgeschlagen, welches eventuell schneller und sparsamer bezüglich des Zwischenspeichers ist.

Dann wurde das Interface-Modell der Gruppe SFU-Interface angesprochen. Hier wurde darauf hingewiesen, daß die bisher favorisierte Lösung mit Ausnutzen der freien Befehle nicht möglich ist. Probleme ergeben sich beim Load/Store Befehl für die FPU. Dieser Befehl ist vom Format her ein IU-Befehl, bei dem der FPU-Opcode ignoriert wird. es bieten sich somit zwei Lösungsmöglichkeiten an.

Erste Möglichkeit wäre einen Befehl zum Umschalten zwischen der FPU und der SFU zu benutzen. Hierbei stellt sich das Problem des Kontextswitches. Bei einem Kontextswitch können wir nicht sichern, welcher Coprozessor gerade aktiv war. Dieses Problem kann nur umgangen werden, wenn die gesamte Spline-Berechnung atomar wäre, was aber kaum zu realisieren ist.

Als zweite Möglichkeit bietet sich an, daß die SFU eigene Befehle erkennt. Hierzu muß, eventuell mit Hilfe eines Filters, sichergestellt werden, daß kein „normaler" Befehl diesen Op-Code in den normalerweise ignorierten Bereich schreibt.
Bei einem Kontextswitch treten bei dieser Lösung auch keine größeren Probleme auf. Die SFU rechnet

eigenständig weiter, bis sie an einem Punkt angelangt ist, an dem sie auf die IU wartet. Bei dem Wechsel zurück in den alten Kontext kann sie sofort weiterarbeiten. Es muß allerdings sichergestellt werden, daß keine verschiedenen Prozeße aktiv sind, welche die SFU benutzen wollen, da nur jeweils eine Berechnung erfolgen kann, und bei einem Kontextswitch zwischen zwei verschiedenen SFU-Benutzenden Prozeßen nicht unterschieden werden kann.

Als letztes wurden die Termine bis zum Vorlesungsende (09.02.1995) geplant.

Gruppe Rechenwerke
Gebraucht werden verschiedene Rechenwerke

- float - Multiplizierer
- float - Dividierer
- int - Addierer
- float - Subtrahierer
- float - Addierer
- float - Vergleicher

Die verschiedenen Rechenwerke sollen in VHDL beschrieben werden. In einer ersten Phase als eine Beschreibung des Kerns, wobei der Hauptgesichtspunkt auf dem Verfahren allgemein liegt, und in einer zweiten Phase als Beschreibung eines float-Rechenwerkes speziell nach den Anforderungen für die SFU. Beide Entwicklungsschienen sollen jeweils simuliert und synthetisiert werden.
In der Zeit bis zum 09.02. soll sich die Gruppe mit der Realisierung des float-Addierers beschäftigen. Danach soll eine Aufteilung in eine Teilgruppe für den Multiplizierer und eine Teilgruppe für den Dividierer erfolgen.

Gruppe Algorithmusentwurf
Die Gruppe Algorithmusentwurf soll bis zum 26.01. die Einarbeitung in Speedchart abgeschlossen haben und sich auf den Lösungsalgorithmus festlegen.
Bis zum 02.02. soll der Algorithmus weitgehend umgesetzt sein und bis zum 09.02. (Vorlesungsende) soll der Algorithmus in Speedchart spezifiziert sein.

Gruppe SFU-Interface
Die Gruppe SFU-Interface soll am 26.01. ein Interface-Konzept als Diskussionsgrundlage vorstellen. In der darauffolgenden Zeit soll mit der Simulation des Interface von Wolfram Hardt begonnen werden. Bis zum 02.02. sollen erste Simulationen durchgeführt werden, und bis zum 09.02. soll die Simulation abgeschlossen sein.