

Die Tricks von MMX

Für Intel ist MMX die bedeutendste Erweiterung seiner Prozessorarchitektur seit dem 386er. Die künftigen Pentium- und Pentium-Pro-Prozessoren sollen mit ihrer Hilfe Multimedia-Anwendungen beschleunigen.

MMX macht dem Anwender so etwa die gleichen Versprechungen wie jede neue Generation von Grafikkarten: Multimedias bunte Bilderwelt soll sich wiederum etwas schneller bewegen. Der Leistungsschub, der bei typischen Operationen der Bildverarbeitung bis zu vierfache Beschleunigung verheißt, geht allerdings ausschließlich vom Prozessor aus – sofern das Programm den neuen Befehlssatz schon nutzt.

Erster Prozessor mit MMX wird der unter dem Code P55C entwickelte neue Pentium sein, der Ende des Jahres erwartet wird. Und auch die Klonhersteller haben bereits ihre MMX-kompatiblen Prozessoren angekündigt: Sowohl AMD als auch Cyrix wollen mit neuen Chips aufwarten. Ein Nachteil von P55C und MMX: Der neue Prozessor verlangt nach einer neuen Hauptplatine.

Bevor der Befehlssatz der Intel-Architektur um 57 spezielle MMX-Befehle erweitert wurde, haben die Entwickler eine Reihe von Applikationen untersucht, darunter Grafik, MPEG-Video, Klangsynthese, Sprachkompression, Bildverarbeitung und Spiele. Das Ergebnis: Bei diesen Programmen tauchen häufig die gleichen rechenintensiven Routinen auf. Deren Kennzeichen sind unter anderem kleine Integerdaten, wie sie für 8-Bit-Bildpixel oder 16-Bit-Audio-Samples gebraucht werden. Weiterhin kommen kleine, sehr oft durchlaufene Schleifen und hochparallele Operationen häufig vor.

Basis von MMX ist die sogenannte SIMD-Technik (Single Instruction, Mul-

tipple Data). Mit ihrer Unterstützung können mehrere Informationen mit einem einzigen Befehl parallel verarbeitet werden. Dazu wurden neue Datentypen eingeführt: gepackte Integerformate, die jeweils 64 Bit groß sind und mehrere Einzeldaten enthalten. Ein packed Byte enthält 8 Byte, ein packed Word vier 16-Bit-Words und ein packed Doubleword zwei 32-Bit-Doublewords. Das Format Quadword (64 Bit) wird ebenfalls unterstützt.

Bei Grafikdaten ist MMX beispielsweise in der Lage, die Daten von acht Pixeln gleichzeitig in ein Register zu schreiben und zu verarbeiten. Acht 64-Bit-Register stehen MMX zur Verfügung. Dabei handelt es sich nicht um neue physikalische Register; sie werden vielmehr von den vorhandenen Gleitkomma-regi-

Die MMX-Befehle umfassen verschiedene Sorten von Funktionen. Dazu gehören Grundoperationen wie Addition, Subtraktion, Multiplikation, arithmetische Verschiebung und Multiplikation /Addition. Enthalten sind ferner Vergleichsoperationen, Konvertierungen zwischen den Datentypen, logische Operationen wie AND, AND NOT, OR und XOR, [Shift]-Befehle sowie Anweisungen zum Datentransport zwischen den Registern oder zwischen Registern und Arbeitsspeicher.

Wichtige Operationen

Im Fall der Addition kann per MMX eine Sättigungsarithmetik zur Anwendung kommen. Das Beispiel zeigt die parallele Addition von Daten des Typs packed Word. Die rechte der vier Additionen liefert einen Wert, der nicht mehr in 16 Bit paßt. Dem Ergebnis wird daher der höchstmögliche Wert FFFFh zugewiesen (analog bei der Subtraktion: 0x0000). Eine Pixeloperation würde in diesem Fall als Sättigungswert die Farbe Schwarz ergeben (bei Subtraktion Weiß).

Bei der Bildbearbeitung kommt es oft nicht mehr auf das exakte Resultat einer Berechnung an, wenn erst einmal ein Sättigungswert erreicht ist; eine schwarze Fläche etwa wird durch einen Schatten nicht noch schwärzer.

Die Operation »packed multiply add« (PMADD) zum Multiplizieren und Speichern ist ein Schlüsselbefehl für Vektormultiplikation, Matrixmultiplikation und verschiedene Filter. Ausgehend von 16-Bit-Daten werden mit PMADD 32-Bit-Daten generiert. Die 16-Bit-Werte werden zunächst zu 32-Bit-Produkten multipliziert, dann paarweise addiert und in zwei Register geschrieben.

Das Ergebnis eines Parallelvergleichs (»packed compare«) kann als Maske dienen, um Elemente aus verschiedenen Inputs zu selektieren. Das verhindert aufwendigere Verzweigungen, die vor allem beim Pipelining in hochentwickelten Prozessoren viel Zeit kosten.

Mit dem »pack«-Befehl lassen sich vier 32-Bit-Werte als 16-Bit-Daten packen, wobei die Sättigungsarithmetik angewandt wird, wenn die Ausgangsdaten zu groß sind. Die entgegengesetzte Anweisung – »unpack« – steht ebenfalls zur Verfügung.

Sättigungs-Arithmetik

a3	a2	a1	F000h
+	+	+	+
b3	b2	b1	3F000h
a3+b3	a2+b2	a1+b1	FFFFh

CHIP

Multiplikation+Addition

a3	a2	a1	a0
*	*	*	*
b3	b2	b1	b0
a3*b3+a2*b2	a1*b1+a0*b0		

CHIP

Parallelvergleich

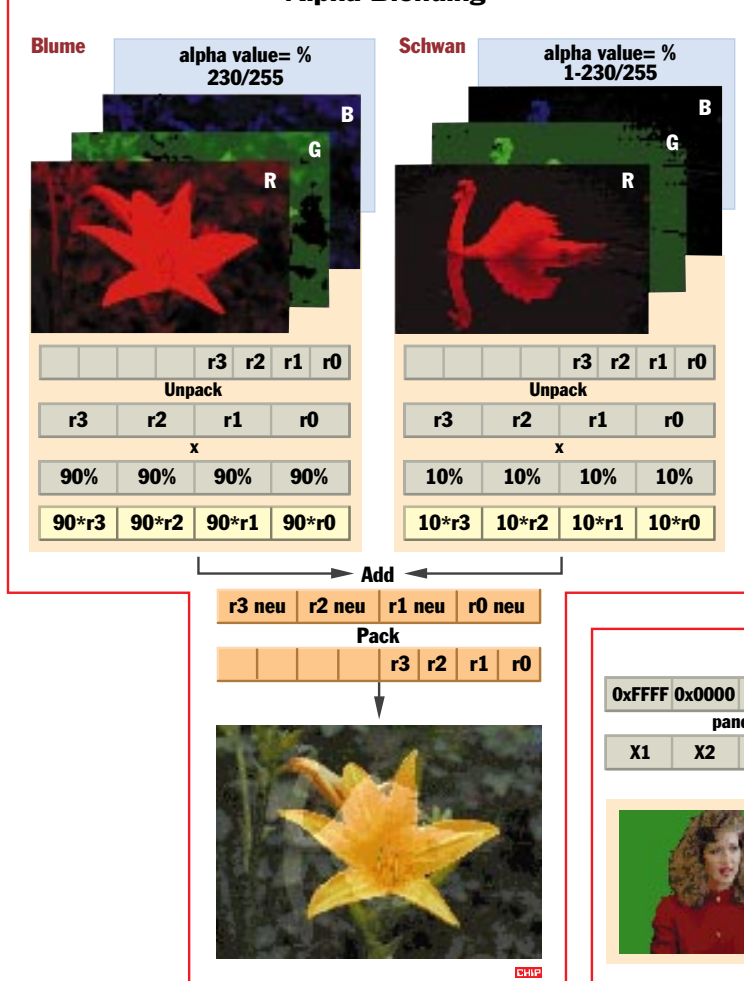
23	45	16	34
a > b?	a > b?	a > b?	a > b?
31	7	16	67
0000h	FFFFh	0000h	0000h

CHIP

Vier auf einen Streich: Mit den MMX-Befehlen können jeweils vier 16-Bit-Daten parallel bearbeitet werden (Erläuterung der Beispiele siehe Text)

stern abgezweigt. Bestehende Betriebssysteme und Anwendungsprogramme merken nichts von den neuen Funktionen und arbeiten wie gewohnt (Kontextwechsel mit Registerspeicherung).

Alpha Blending



funktioniert das ebensoschnell wie 8-Bit-Grafik. Der Trick: Für Bilder mit 24 oder 32 Bit Farbtiefe werden die Farben Rot Grün und Blau jeweils als 8-Bit-Wert dargestellt. Weitere 8 Bit geben für 32-Bit-Farben den sogenannten Alpha-Wert an, der mit seiner Skala von 0 bis 255 die Farbintensität bestimmt.

Ein Anwendungsbeispiel dafür ist das Überblenden von Bildern, das mit einer Farbtiefe von maximal 24 Bit möglich ist. In unserer Grafik (links) werden die Bilder einer Blume und eines Schwans zunächst in ihre Grundfarben zerlegt und dann übereinandergelagert.

Beim Mischen, dem sogenannten Alpha Blending, bestimmt der Alpha-Wert die Intensität der beiden Bildanteile. Wird für die Blume der Alpha-Wert 230 gewählt, so zeigt das Endergebnis 90 Prozent Blume und 10 Prozent Schwan. Die

Chroma Keying



Doch grau ist alle Theorie, und Multimedia's grüner Baum treibt bunte Blüten. Wie sieht der Pakt mit MMX nun in der Praxis aus?

Das sogenannte Chroma Keying ist eine Overlaytechnik, die verwendet wird, um Texte über einen Grafik- oder Videohintergrund zu legen oder um Figuren in einem Spiel agieren zu lassen. Es ist das gleiche Prinzip, mit dem im Fernsehen der Meteorologe vor die Wetterkarte gestellt wird. In unserem Beispiel (siehe Grafik rechts) dient ein grüner Hintergrund dazu, das Bild einer Frau vor Frühlingsblüten zu setzen. Im Beispiel werden vier 16-Bit-Pixel parallel verarbeitet.

Zunächst werden vier Pixel von dem Bild mit der Frau auf grünem Hintergrund genommen. Die erste Datenreihe zeigt abwechselnd die Pixel Grün, NOT Grün, Grün, NOT Grün. Der »Compare«-Befehl entwirft eine Maske für diese Daten, denen nach der Booleschen Arithmetik die logischen Werte 1 und 0 (wahr oder falsch) zugewiesen werden. Der unerwünschte Hintergrund unterscheidet sich so von demjenigen Bildteil,

der behalten werden soll und in der Grafik als Schattenriß dargestellt ist.

Die entstandene Maske wird nun auf das Bild mit den Frühlingsblüten gelegt. Hier sieht man die gleichen vier Pixel wie im anderen Bild. Die Befehle »PANDN« (gepacktes »und nicht«) sowie »PAND« (gepacktes »und«) bestimmen darüber, wo Pixel des Blütenhintergrunds und der Frau erhalten bleiben. Unerwünschte Pixel werden auch im zweiten Bild auf 0 gesetzt. Das endgültige Bild baut der »POR«-Befehl (gepacktes »oder«) auf.

Vier Pixel wurden so mit vier MMX-Befehlen gemappt, ohne daß eine Verzweigung erfolgte. Ohne MMX müßte jedes Pixel einzeln verarbeitet werden und eine eigene Abfrage erfordern.

MMX bietet die Möglichkeit, von 8-Bit- oder 16-Bit-Grafik auf 24-Bit-Grafik oder sogar Echtfarben (32 Bit) überzugehen, was beispielsweise Spiele realistischer erscheinen läßt. In vielen Fällen

Farben werden dabei nacheinander verarbeitet, zunächst je vier rote Pixel von Blume und Schwan (siehe Grafik), dann die grünen und schließlich die blauen.

Da sich der Alpha-Wert von Bild zu Bild variieren läßt, wird diese Technik gerne von Spieleentwicklern benutzt. Staub- und Rauchwolken oder der Fisch im Wasser wirken damit realistischer.

Manfred Flohr

Internet-Adresse:
<http://www.intel.com/pc-supply/multimed/mmx/index.htm>