

Digitale Signal-Prozessoren - DSP Teil 1

Grundlagen und praktische Anwendungen von DSPs

Diese Artikelserie vermittelt die Grundlagen der speziell für die digitale Signalverarbeitung konzipierten Prozessoren. Eine durch umfangreiche Software unterstützte PC-Einsteckkarte mit verschiedenen Erweiterungen bietet eine Vielzahl neuer Experimentier- und Anwendungsmöglichkeiten.

1. Elektronische Signalverarbeitung

Allgemein versteht man unter elektronischer Signalverarbeitung

- die Manipulation von Signalen, d. h. das Herausfiltern von Informationen aus einem Signalgemisch
- das Generieren von gewünschten Wellenformen
- die Modifikation der Amplitudencharakteristik eines Ausgangssignals oder
- das Herausfiltern von unerwünschten Komponenten usw.

In der Vergangenheit wurden derartige Signalveränderungen mit analogen Schaltungen vorgenommen, die jedoch mit einer ganzen Reihe von Problemen, wie z. B. Genauigkeit, Alterung, Temperaturabhängigkeit und Drift behaftet sind. Der Schaltungsabgleich ist zudem bei analogen Konzepten häufig mit großem Aufwand verbunden.

Für die digitale Verarbeitung der mei-

sten analogen Signale reicht jedoch die Verarbeitungsgeschwindigkeit herkömmlicher Mikrocontroller bei weitem nicht aus.

2. Der digitale Signalprozessor - eine experimentelle Herausforderung

Durch die Aufgabenstellung, analoge Signale digital zu bearbeiten, wurde eine Prozessorstruktur entwickelt, welche 1979 zum ersten programmierbaren Spezialprozessor für dieses Aufgabengebiet führte und auch seinen Namen - Digitaler Signal Prozessor - prägte.

Aufgrund der für diesen Problemkreis typischen Aufgaben der Transformation (z. B. Fourier-Transformation) werden an diese Prozessoren vor allem Anforderungen in Rechengeschwindigkeit und Datendurchsatz gestellt. Dies ist um so klarer, da analoge Signale, wie zum Beispiel Audiosignale, immer in Echtzeit vorliegen und auch damit in Echtzeit ver- oder bearbeitet werden müssen.

Zunächst war die Verarbeitung von Audiodaten das bevorzugte Einsatzgebiet der Signalprozessoren. Erst gegen Ende der 80er Jahre kamen digitale Signalprozessoren zunehmend auch in anderen Anwendungen zum Einsatz, wo ihre Eigenschaft der Echtzeitverarbeitung genutzt werden konnte.

Auch die positive Preisentwicklung gestattet es zunehmend, diesen Prozessor als leistungsstarke Verarbeitungseinheit in allgemeinen Anwendungen der Meß- und Regelungstechnik einzusetzen. Inzwischen kommen zur Audiosignalverarbeitung Aufgaben wie Bildverarbeitung und Protokollkonvertierungen in der Kommunikationstechnik hinzu, welche die hohen Datendurchsätze und Rechenleistungen nutzen. Dabei bewegt sich heute das aktuelle Leistungsspektrum von einigen MOPS (Millionen Operationen pro Sekunde) eines TMS320C10 bis zu 2000 MOPS beim TMS320C80.

Signalprozessoren werden nach den Eigenschaften der Arithmetikeinheit (Gleit-

komma und Integer) und nach der Datenbreite (16, 24 oder 32 Bit) unterschieden.

Diese Artikelserie wird sich u. a. mit einer PC-Karte befassen, die neben einer experimentellen Arbeit auch technische Einsätze mit dem DSP erlaubt. Wir haben dafür einen einfachen, aber leistungsfähigen Vertreter mit 16 Bit Verarbeitungsbreite und Integerarithmetik gewählt.

In folgenden Beiträgen stellen wir verschiedene Zusatzmodule vor, mit denen verschiedene Aufgaben realisierbar sind, angefangen bei einer (einfachen) Bildverarbeitung bis hin zu zahlreichen Projekten aus der Steuerungs- und Regelungstechnik.

Mit diesen Erweiterungsbausteinen ist die ELV-DSP-Karte direkt für unterschiedliche Meß- und Steuerungsaufgaben einsetzbar.

2.1. Die Leistungsmerkmale eines DSP

Die bereits angesprochene hohe Verarbeitungsleistung wird bestimmt durch eine Reihe konstruktiver Eigenschaften des DSP. Als wesentliche sind dabei zu nennen:

- geringe Befehlsausführungszeiten
- ein Rechenwerk (ALU) mit komplexen Funktionen, wie Multiplikation und akkumulierende Multiplikation
- mehrere unabhängige Busse für einen hohen Datendurchsatz
- einen optimierten Befehlssatz
- paralleler Ablauf mehrere Funktionen
- große interne Speicherbereiche

Die geringen Befehlsausführungszeiten erreicht man durch eine den RISC-Prozessoren ähnliche Architektur, wobei einfache Befehle mit hoher Geschwindigkeit, allgemein in einem Taktzyklus, abgearbeitet werden.

Da dies technisch schwer realisierbar ist, werden die Befehle in Einzelaktivitäten (wie Befehl lesen, Befehl decodieren, Operanden holen, ...) zerlegt.

Die Befehle durchlaufen nun diese Arbeitsstufen, so daß mit jedem Takt ein neuer Befehl aufgerufen und die Bearbeitung eines zuvor gelesenen Befehls abgeschlossen wird. Allgemein bezeichnet man dieses Verfahren als Pipelining. Auf diese Weise ist es dann auch möglich, daß mit jedem Takt eine Multiplikation mit anschließender Addition des Ergebnisses zum Akkumulator erfolgen kann.

Um die erforderlichen Datenmengen zu bewältigen, nutzen DSPs nicht den bei Mikroprozessoren üblichen einfachen Bus für Daten und Programm (von-Neumann-Architektur), sondern besitzen für Daten und Programm getrennte vollständige Bussysteme (Harvard-Architektur). Damit ist es z. B. möglich, den nächsten Befehl zu holen, während das Ergebnis der vorheri-

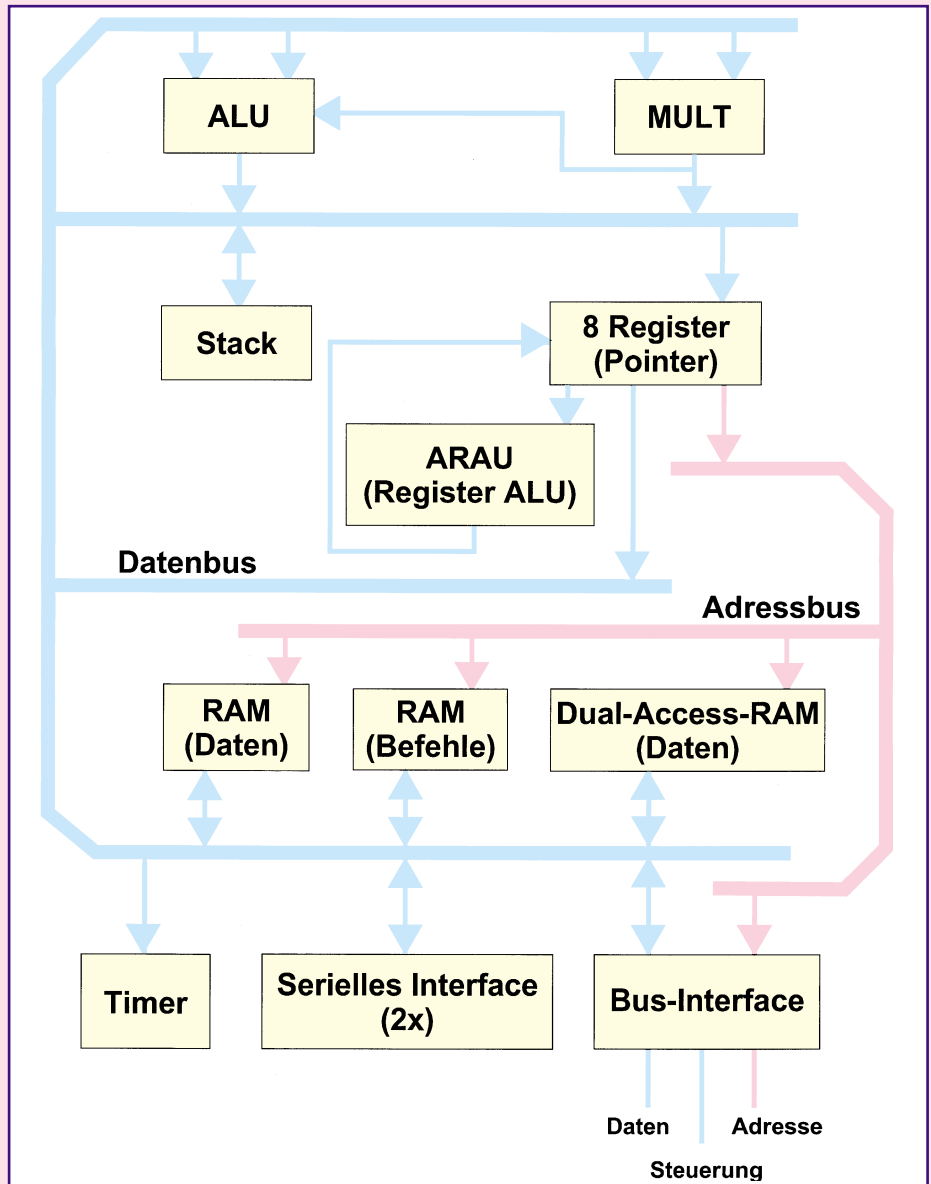


Bild 1: Vereinfachtes Blockschaltbild des DSP TMS320C50

gen Operation im Datenspeicher abgelegt wird.

Da für eine Operation außer dem Befehl meist zwei Operanden benötigt werden, realisiert man im Prozessor mit dem integrierten RAM meist außer dem Befehlsbus zwei unabhängige Datenbusse (erweiterte Harvard-Architektur), so daß man zur gleichen Zeit zwei Operanden lesen oder je einen lesen und schreiben kann.

Da jedoch 3 Bussysteme außerhalb des Prozessors einen hohen technischen Aufwand bedeuten und eine unerwünschte Quelle bzw. Empfänger von Störstrahlung darstellen, beschränkt man die externe Beschaltung auf 2 Bussysteme oder ein Bussystem mit getrennten Steuersignalen für Daten und Programm (modifizierte Harvard-Architektur).

Hinzu kommt, das externe Bussysteme durch Leitungseigenschaften und Signallaufzeiten von Treiber- und Decoderbausteinen langsamer sind als im Chip inte-

grierte Speichereinheiten. Aus diesem Grund vergrößert man ständig die auf dem Chip befindliche Speicherkapazität, um die optimale interne Busstruktur nutzen zu können. Die derzeit üblichen Speicherbereiche bewegen sich von 1 kByte bis zu 500 kByte, was auch einen maßgeblichen Einfluß auf den Preis hat.

Der letzte hier zu betrachtende Punkt ist die Realisierung paralleler Abläufe. Dabei ist es üblich, einen vom Rechenwerk unabhängigen Hardware-Multiplizierer zu integrieren. Zusätzlich wird ein zweites, einfacheres Rechenwerk integriert, welches für die Pointer-Arithmetik zuständig ist. Dadurch können Manipulationen an den Operandenadressen unabhängig und zeitgleich zur eigentlichen Operation ausgeführt werden.

In Abbildung 1 sind die wesentlichen Komponenten eines DSP dargestellt. Diese vereinfachte Darstellung wird durch eine große Anzahl von Steuerregistern und Mul-

tiplexern ergänzt, zur Organisation des Datenflusses. Bei dem in Abbildung 1 gezeigten Digital-Signalprozessor oder DSP handelt es sich um den Typ TMS320C50, mit dem wir uns im weiteren detailliert befassen wollen.

Der Signalprozessor ist durch Baugruppen, wie Timer, serielle Schnittstelle und Interruptcontroller, ergänzt, die erst den Einsatz als eigenständigen Rechner ermöglichen.

2.2. Warum nehmen wir keinen Mikrocontroller?

Wir hatten zwar schon festgestellt, daß der DSP geeignet ist, große Datenmengen zu verarbeiten. Wenn man von den spezifischen Unterschieden in Hard- und Software absieht, kann man einen TMS320C50 mit einem 80C51 vergleichen. Beide besitzen interne Speicherbereiche, Timer sowie ein serielles Interface und sind für Taktfrequenzen bis 40 MHz verfügbar.

Der Hauptunterschied liegt in der Methode der Befehlsausführung - ein TMS320C50 benötigt einen Takt, ein 80C51 dagegen 12 Takte für einen Befehl. Da der 80C51 üblicherweise vom externen ROM arbeitet, werden praktisch Prozessoren mit 16- bzw. 24MHz-Takt eingesetzt, so daß ein Programm um den Faktor 20 bis 30 langsamer wird. Dazu kommt noch die Struktur der Befehle, die auf die Geschwindigkeit ebenfalls einen wesentlichen Einfluß nimmt. Das soll ein kleines Beispiel verdeutlichen:

Wir wollen den Mittelwert aus 8 Werten einer Tabelle (im internen RAM) berechnen. Mit dem TMS320C50 hat das Programm das in Tabelle 1 gezeigte Aussehen. Zum Vergleich wird in Tabelle 2 die gleiche Aufgabe mit dem 80C51 dargestellt.

Bei einer Taktfrequenz von 40 MHz ergibt das eine Laufzeit von 39µs. Der 80C51 ist damit um etwa den Faktor 100 langsamer als der DSP. Dieses Verhältnis läßt sich durch moderne Vertreter dieser Familie noch etwas verbessern. So erreicht man mit dem als 12 MHz verfügbaren 80C251 ca. 11µs Laufzeit. Bei dieser Betrachtung ist noch nicht berücksichtigt, daß der DSP eine 16-Bit-Verarbeitung (mit 32-Bit-Akkumulator) durchführt, wogegen beim 80C51-Beispiel nur mit 8 Bit und 16-Bit-Ergebnis gearbeitet wurde.

2.3. Was tun mit so viel Rechenleistung?

Um solche hohen Rechenleistungen sinnvoll einzusetzen, gibt es 2 Möglichkeiten: Aufgaben mit hoher Komplexität oder man ersetzt bei entsprechenden Anwendungen die Hardware durch Software.

Beispiele für eine hohe Komplexität sind Aufgaben der Datenübertragung und Pro-

Tabelle 1: Mittelwert berechnung aus 8 Werten mit dem TMS320C50

			Takte	
anf:	LAR	AR0,#tabelle	; Laden der Anfangsadresse der Tabelle	2
	LARP	AR0	; Arbeitsregister 0 als Speicherpointer	1
	ZAP		; Akku := 0	1
	RPT	#7	; Folgebefehl 8x ausführen	1
	ADD	*+	; Add indirekt Inhalt der Tabelle und erhöhe	8
			; Tabellenzeiger (AR0) um 1	
	BSAR	#2	; Akku 3 Bit rechts schieben (AC:= AC/8)	1

Das ergibt in der Summe 14 Takte * 25ns = 350ns.

Tabelle 2: Mittelwertberechnung aus 8 Werten mit dem TMS80C51

Takte				
	CLR	A	; Löschen A	12
	MOV	R0,#result	; Zeiger auf result	12
	MOV	@R0,A	; result(H) :=0	12
	INC	R0		12
	MOV	@R0,A	; result(L) :=0	12
	MOV	R7,#LOW array	; Pointer laden	12
	MOV	R6,#08H	; Schleifenzähler	12
Loop1:				8 mal
	MOV	R0,AR7	; Pointer Tabelle nach R0	12
	MOV	A,@R0	; Byte nach A	12
	MOV	R0,#result+01H	; Pointer result nach R0	12
	ADD	A,@R0	; A := a + result(L)	12
	MOV	@R0,A	; Ergebnis ablegen	12
	DEC	R0	; Pointer -1	12
	CLR	A	; a :=0	12
	ADDC	A,@R0	; addiere CY + result(H)	12
	MOV	@R0,A		12
	INC	R7	; Pointer + 1	12
	DJNZ	R6,Loop1	; Schleife	24
	MOV	R0,#result	; Pointer auf result(H)	12
	MOV	A,@R0		12
	MOV	R6,A	; r6 := result(H)	12
	INC	R0		12
	MOV	A,@R0	; A := result(L)	12
	MOV	R0,#03H	; Schleifenzähler	12
Loop2:				3 mal
	XCH	A,R6	; A := result(H)	12
	MOV	C,ACC.7	; sign-Extension	12
	RRC	A	; 1 Bit rechts	12
	XCH	A,R6	; A := result(L)	12
	RRC	A	; 1 Bit rechts	12
	DJNZ	R0,Loop2	; Schleife	24
	; Ergebnis steht in A			
			Summe	1560 Takte

tokollkonvertierung. Dazu kommen die Gebiete der Bildverarbeitung, wo zur Geschwindigkeit noch die Datenmenge kommt sowie die Arbeit mit schnellen Sensoren.

Die Verlagerung einfacher Aufgabenstellungen aus der Hardware in eine reine Softwarelösung hat mehrere Vorteile. So ist die Zuverlässigkeit einer Baugruppe umso höher, je weniger Bauteile enthalten sind. Die Softwarelösung besitzt zudem eine höhere Flexibilität, da die „Schaltung“ per Programm geändert werden kann.

Auch unter dem Gesichtspunkt der EMV-Bestimmungen ist der Einsatz eines einzigen, intern arbeitenden Schaltkreises günstiger. Man kommt mit dieser Lösung allerdings auch sehr schnell an die Geschwindigkeitsgrenzen, da eine komplexe logische Operation (mit Ein- und Ausgabe) im Zeitbedarf 1 µs schnell erreicht.

Mit dem in dieser Artikelserie vorge-

stellten DSP-Board (PC-Einsteckkarte) und den Zusatzbaugruppen wollen wir 3 Wege beschreiten:

- Es sollen Beispiele und Algorithmen aus der Bildverarbeitung vorgestellt werden
- wir betrachten Anwendungen der schnellen digitalen Signalauswertung
- und wir realisieren einen schnellen optischen Kommunikationsknoten für Meß- und Steuerungsaufgaben.

Im 2. Teil dieses Artikels folgt die ausführliche Vorstellung und Beschreibung der innovativen Schaltungstechnik des ELV-DSP-Boards, während im 3. Teil Nachbau und Inbetriebnahme detailliert erläutert werden. Im daran anschließenden weiteren Verlauf dieser Artikelserie folgt die Programmierung und die Darstellung verschiedener Zusatzbaugruppen zum DSP-Board, so daß der interessierte Elektroniker in der Lage ist, unterschiedliche Anwendungen zu realisieren.