Weiter DSP

4.2.2. DSP der mittleren Leistungsklasse

Bsp. TMS320… (F48)

- Prozessorkern:
  - Pipeline mit Inst. Fetch , Dispatch (Aufteieln auf die Datenpfade), Dekoder, zwei Datenpfde mit Registersatz (Berechnungen, 2 fach parallel)
- Weitere Funktionseinheiten
  - Befehlscache (1. Level, 2. Level)
  - Datencache (1. Level, 2. Level)
  - Harwardarchitektur bis zu den Caches (getrennter Zugrif auf Befele und Operanden)
  - DMA für Speicherinterface (EMIF, Extended Memory Interface) und EA
  - * z.B Timer, General Purpose INPUT OUTPUT; gpio9 I^2C-Businterface, …

4.2.3. DSP aus dem Hochleistungssegment

Bsp. Tigersharc von Analog Devices (F49)

- Prozessorkern: 2 vollständige Verarbeitungseinheiten mit je parallel normale ALU und Multiplizier- und Addierpfad
- Weiter Funktionseinheiten:
  - 3 parallele Bussysteme mit Datenbreite 128 bit (größere Datenformate sind direkt übertragbar)
  - System on Chip Inetrface (SOC) für Anschluss weiter Funktionseinheiten auf dem Chip
  - Je Verarbeitungseinheit 1 Adressgenarato
  - 1 Befehlssequenzer mit Brach Target Buffer (Sprungvorhersage)

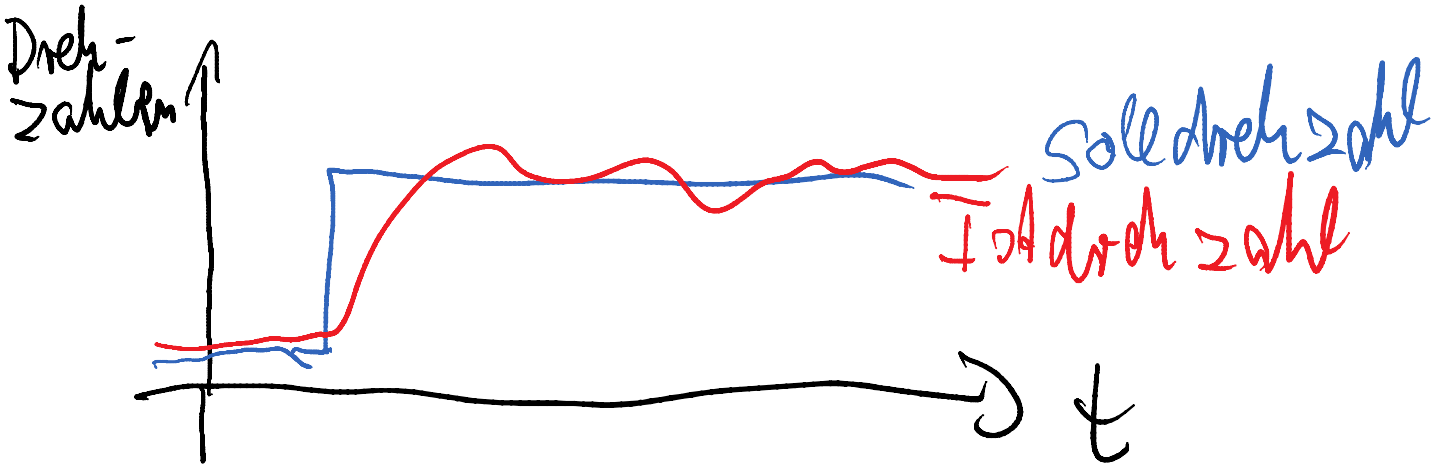
- Interner Speicher mit Kreuzschienenschalter (Crosbar) zum Zugriff auf den Speicher von allen (4) vorhandenen Bussystemen (das 4. ist für externen Speicherzugriff)

- Erweiterungen am SOC (F50)
  - DMA (schneller Zugriff auf die Links (für die Kopplung zu weitern DSPs, evtl. EA-Anschluss) und Hostinterface
  - Hostinterface: Kopplung zu einen Universalrechner (z.B. ein PC)

BSP. für ein DSP (hier Tigersharc) Programm

PID-Regler (Regler mit Proportional-, Integral- und Differtial-Anteil)

Problem hier:
- Regeln einer Motordrehzahl entsprechend vorgegebener Solldrehzahl (r(n))
- u(n) berechnete Stellgröße
- DA-Wandler: Unwandlung von u(n) in eine analoge Spannung zur Motoransteuerung
- AD-Wandler: Umwandlung der analog gemessenen Drehzahl in Digitalwert (Istdrehzahl, y(n)
- e(n) Regelaweichung, der Regler soll diese auf 0 regeln, bei 0 ist Istdrehzahl= Solldrehzahl



Gleichung kontinuierlich:
F51 oben.

Konvertierte diskrete Gleichung:
(F51 unten) Prinzip Integral und Differential und Proportionaanteil werden ein Polinom aus der Sucke von Koeffizientem*Regelabweichung (3 Tastpunkte und Koeffizient*Stellwert letzter Tastpunkt)

Koeffizienten der dikrteten Form errechnen sich aus den Koeffizienten der kontunuierlichen Form und der Tastperiode T.

Effektive Realisierung des Polinoms: -> hohe Motordrehzahlen regelbar

Struktur des Ploinoms als Datenflussplan:
+ Additionen
Pfeile mit Ki sind die Muultiplikationen
z^{-1} Verzögerung um ein T

➔ F52

Sehr wenig Befehle:
Lesen und Schreiben von und auf ein Port (EA)
MAC-Befehle (XMR) für die Berechnung des Polynoms
Umkopieren von Wertenden für das Poynom für die nächste Berechnung nächster Tastpunkt (n->n-1; n-1->n-2)

Ausführungszeit 14 ns -> möglicher Tastfrequenz 1/14 GHz
Höchste im Signalspektrum mögliche Frequenz 1/28 GHz (Abtastheorem von Shannon)

➔ Diese hohe Frequenz hier für den Motor nicht unbedingt notwendig.

5. Parallele Architekturen

Bisher: i.a. Mikroparallelität (Parallelstrukturen im Prozessor, Speicher usw.) (außer im DSP zwei vollständige Recheneinheiten)

In Kap. 5 i.a. Makroparalleleität, d.h. Prozessorkerne parallel.

Klassifikation nach Flynn (F54):

- Zusammenhang Befehlstrom, Datenstrom

- SISD (F55): Single Instruction, Single Data

  - Ein Befehlsstrom wirkt auf einen Datenstrom (normaler Ein-Kern-Prozessor)

- SIMD (F56) Single Instruction Multible Data
  - Ein Befehl wirkt in mehreren Recheneinheiten auf mehrere Daten (z.B. für eine Vektoroperatin für mehrere Elemente gleichzeitig, Bsp. Vektorrechner, MMX (Multimedia Extension in PC-Prozessoren; Mikroparallelität)

- MISD (F57) Multible Instruktion, Single data, mehrere Befehle wirken gleichzeitig auf selben Daten, wenig Bedeutung im praktischen Einsatz)

- MIMD (Multible Instruction Multible Data) Mehrere Befehle wirken gleichzeitig auf mehrer Datn) z.B Multicore architekturen, Parallelrechner, Rechnernetze (hier unterste Hierrchieebene), wichtigste Klasse