

RECHNERARCHITEKTUR

AVR-Microcontroller

Bausteine

ALU - Arithmetic Logic Unit (Funktionsgenerator)	Ausführen von mathematischen Funktionen
Register File (Register)	32 Register je 8bit zum Speichern Die letzten sechs sind die (X,Y,Z)-Register
Instruction Register (Anweisungen)	Beinhaltet aktuelle Instruktion
Program Memory (Programm speicher)	wandelt code in bytes im add r17,r16 → 00011110001000
Sequencer (Programm counter)	besteht aus dem program counter (pc) Stand des abzulaufenden Prozesses
Status Register (Statusregister)	speichert carry, zero, negativ, ... -flag
IO Port (Ein-/Ausgabe)	Daten ein- und ausgeben jeder PIN kann individuell eingestellt werden
	PORTX: heraus DDRX: schreiben 1 / lesen 0 PINX: hinein
RAM (Arbeitsspeicher)	größer als Programm Memory schnell, keine direkten arithmetischen Operationen muss mit einem Offset angesprochen werden (\$0100)
	beinhaltet System Stack Last In - First Out für Subroutine und Interrupt - sichert PC und Statusreg. spezielles Register, da verschachtelte Subroutinen
Interrupt Unit (Unterbrechungen)	acht verschiedene Interrupts (INT0 .. INT7)

Begriffe

Der **Instruktionsdekoder** befindet sich der Übersichtlichkeit halber her im jeweiligen Baustein.

Harvard-Architektur: Programmspeicher und Datenspeicher sind separiert. Programmspeicher = ROM

instruction fetching: Holen von Instruktionen aus dem Programmspeicher und laden ins Instruktionsregister

IO = Zustandsautomat, keine logische Schaltung

distributed ram: - RAM benutzt ebenfalls normale D-Register

tos = Top-of-Stack

nrst = Reset

clk = Clock

carry = Übertrag

reg = data = Register

Pipeline = Test für parallel ablaufende Prozesse

bei Alu = MAC (Multiplie and accumulate) 2stufig

Spezialitäten des DSP

- Zero-Overhead Loops
- MAC (Multiplie and accumulate)
- Adress generation units

MATLAB

```
[b,a]=butter(2, 1000/24000);
```

```
freqz(b,a,512,48000)
```