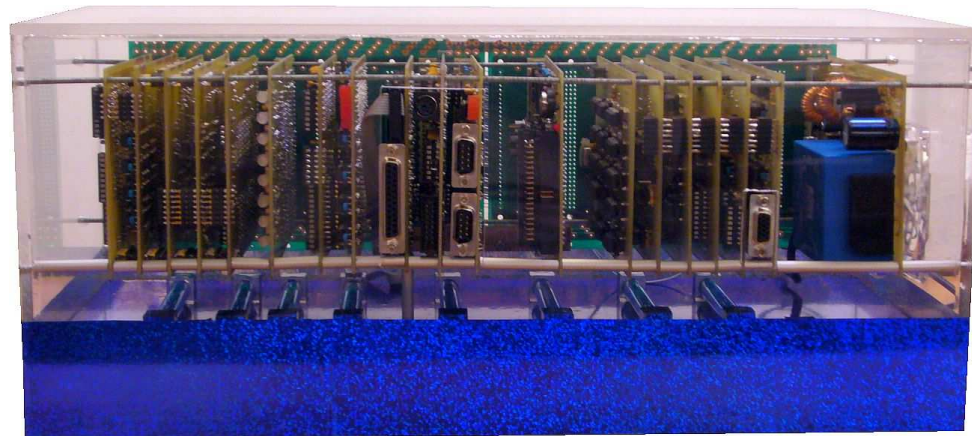


[www.MyCPU.eu](http://www.MyCPU.eu)



Ein Computer zum Selbstbauen

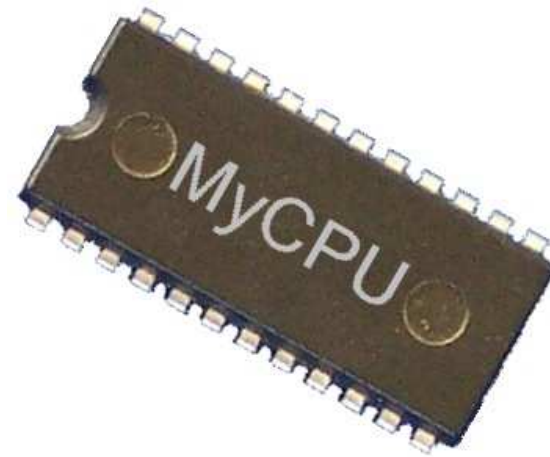
von Dennis Kuschel

## Gliederung:

1. Was ist die MyCPU und warum existiert sie ?
2. Architektur der MyCPU
3. Befehlssatz
4. Emulator
5. Software
6. Nachbau

# Was ist die MyCPU ?

- 8 Bit Prozessor
- max. 8 MHz „schnell“
- Harvard Architektur
- 64k Programm- und 64k Datenspeicher adressierbar
- komplett aus Einzelteilen aufgebaut
- besteht aus 61 IC's der 74er Serie auf 5 Platinen
- verwendet keine Spezial-IC's
- kann von jedermann nachgebaut werden
- mit weiteren Komponenten entsteht ein kompletter „PC“





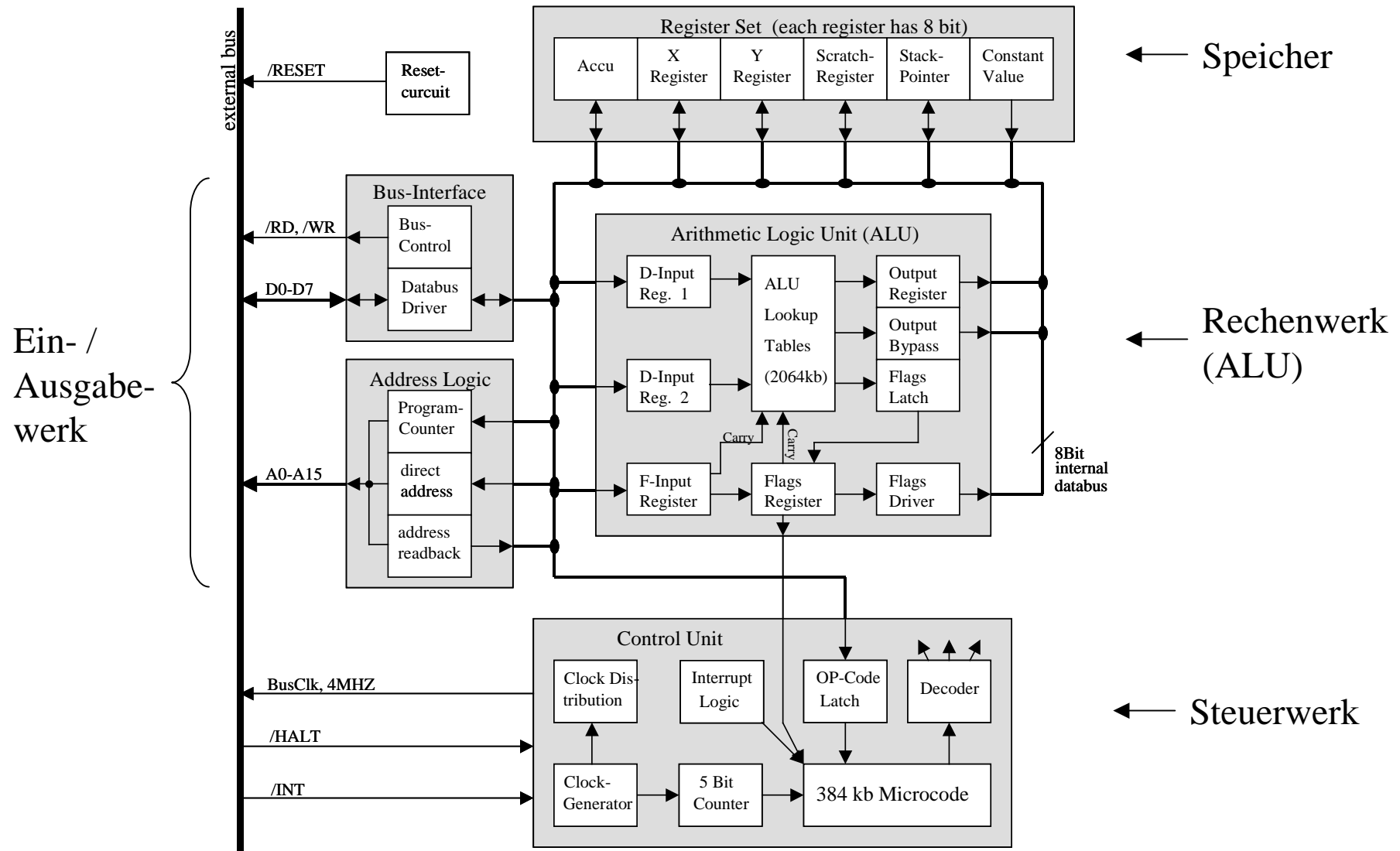
MyCPU: 5 Platinen, 61 IC's

# Warum das Ganze?

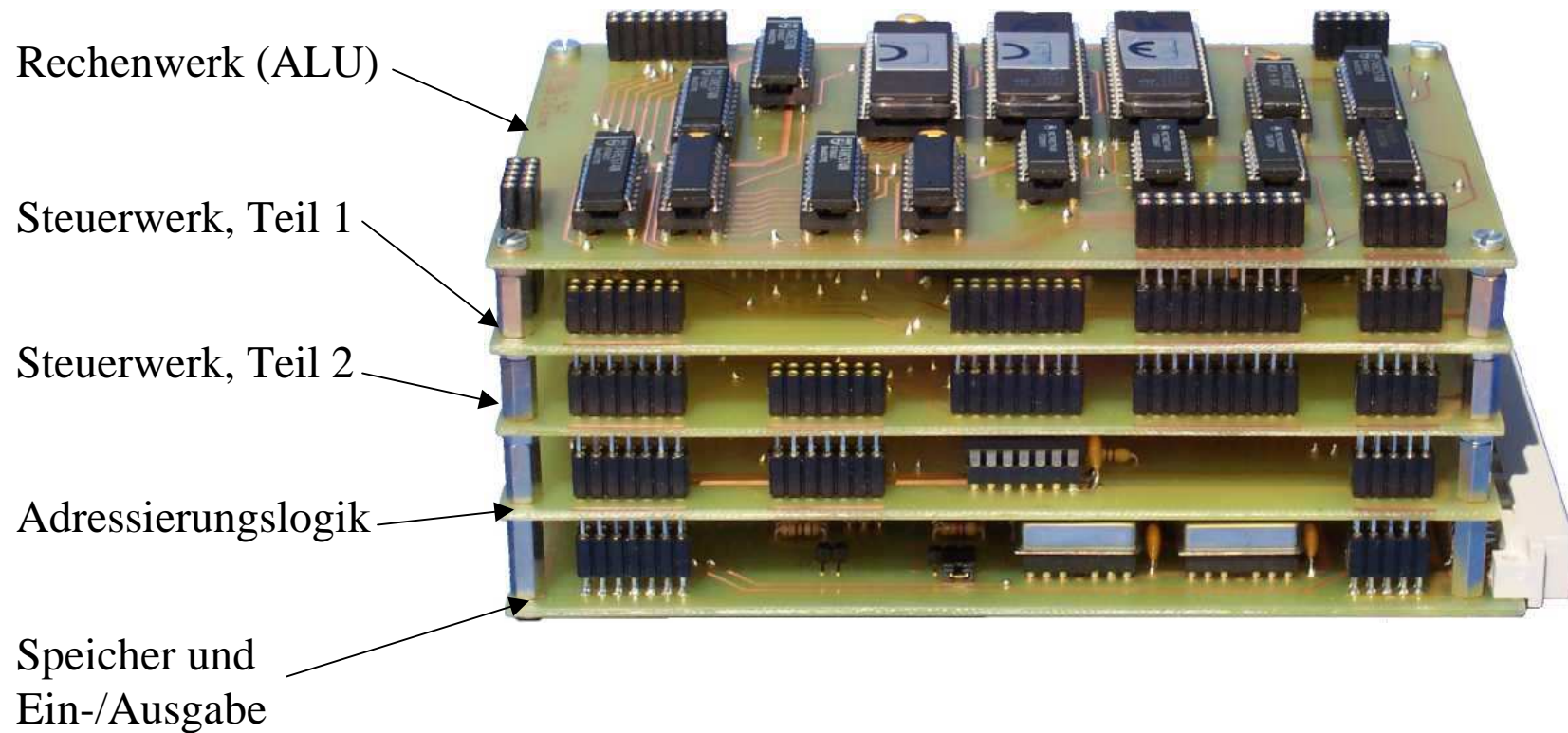
- Die MyCPU entstand aus einer Laune heraus. Während der Vorbereitungen zu meiner Diplomprüfung hatte ich Langeweile.
- Inspiriert wurde ich durch die Vorlesung „Digitaltechnik“
- Die MyCPU sollte anfangs nur ein Taschenrechner werden
- Absolute kreative Freiheit (Architektur, Software, u.s.w.)
- Der Weg ist das Ziel. Es gibt viel zu lernen.
- Andere können hoffentlich etwas von meiner CPU lernen

„Andere haben eine Modelleisenbahn im Keller, ich eine CPU !“

# Aufbau der MyCPU

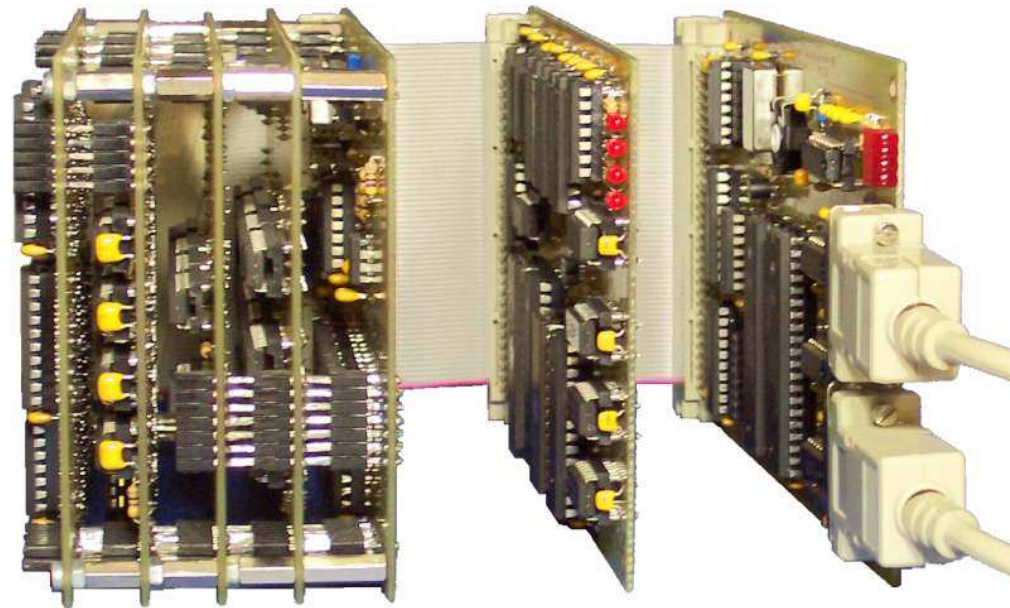


# Aufbau der MyCPU auf 5 Platinen





# Ein kleiner Computer mit der MyCPU



Zentraleinheit  
(MyCPU)

Daten- und  
Programmspeicher  
(Permanent: ROM)

Ein- und Ausgabe über  
ein Terminal via RS232  
(kann ein PC sein)

Bauteilkosten: 400 €, Arbeitsaufwand: Je nach dem... Basteln macht doch Spaß!



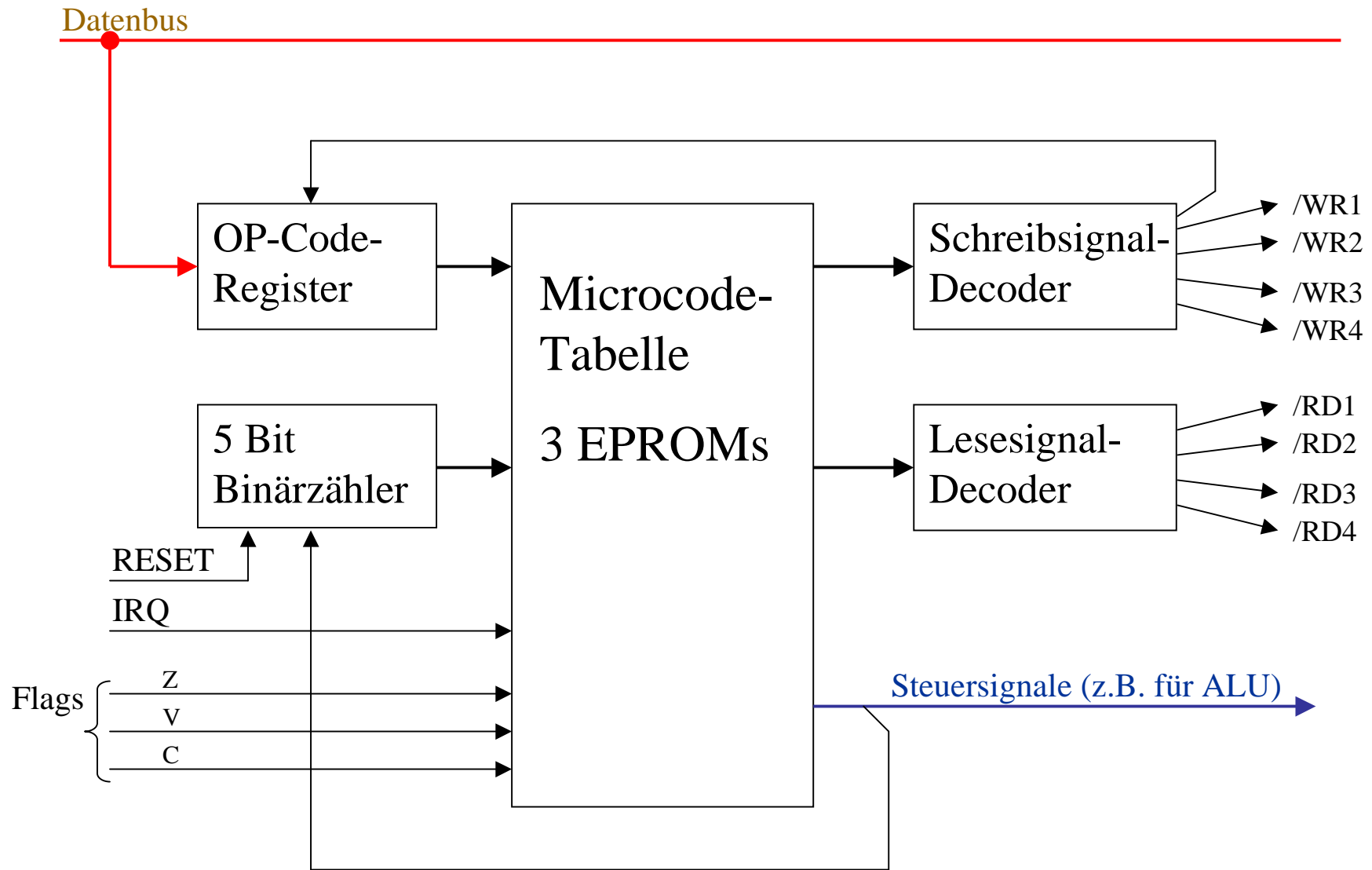
# Ein echter Computer mit der MyCPU !



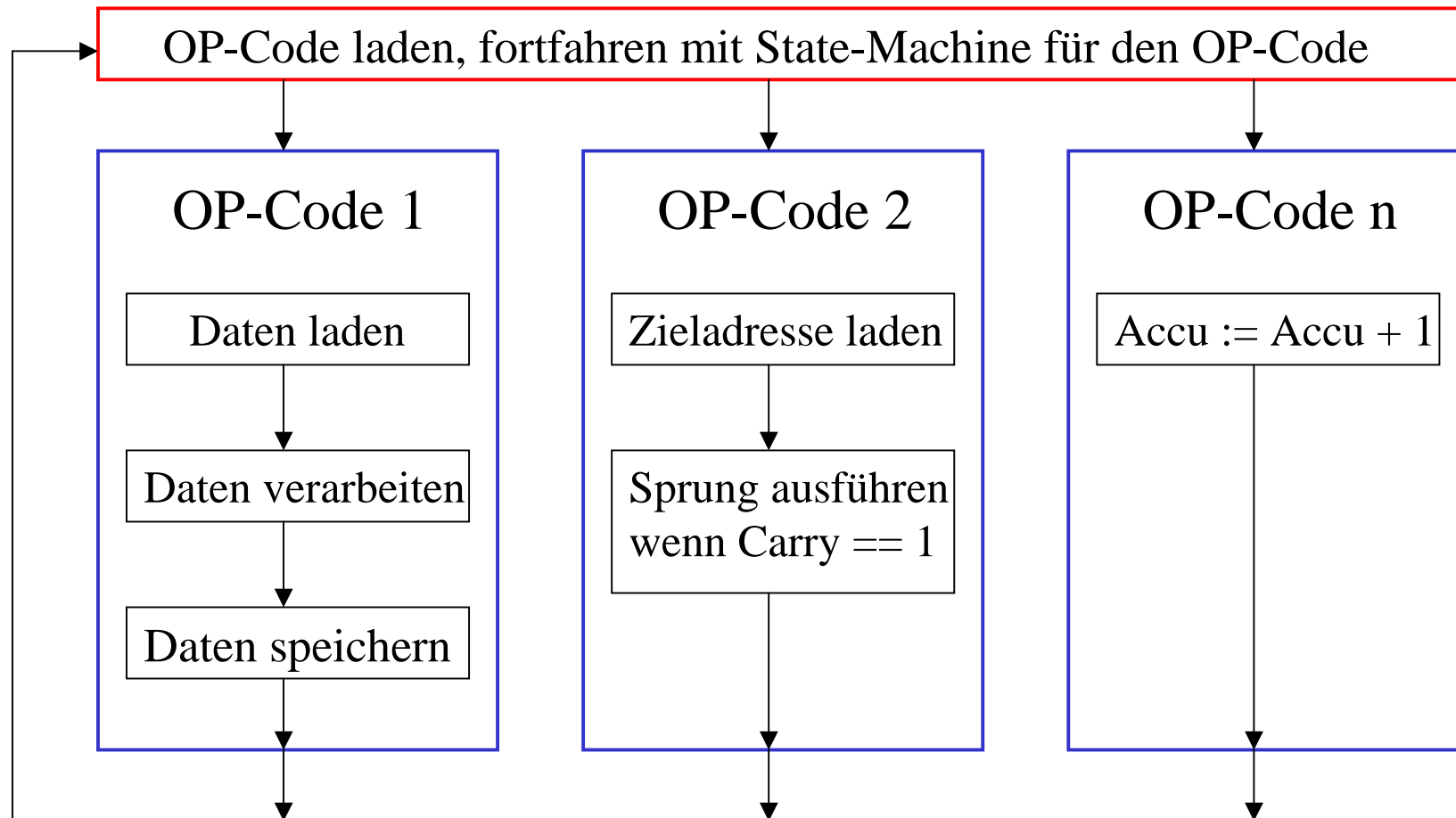
8 MHz CPU, 1216 kB Speicher,  
Festplatte, Tastatur, Grafikkarte



# Aufbau des Steuerwerks



# Ablaufsteuerung (State Machine)



⇒ Innerhalb eines OP-Codes kann nicht zurück gesprungen werden.  
Das Überspringen von Mikro-OP's ist allerdings möglich.

# Der Mikrocode Compiler

- Für die Erstellung der 256 Mikrocodes wurde ein spezieller Compiler geschrieben.
- Ein Mikrocode kann komfortabel nach dem Schema „Lade OP-Code, lade Operand vom RAM in ALU, lade Inhalt von Akku in ALU, führe Addition aus, speichere das Ergebnis der Addition im Akku“ erstellt werden.
- Der Compiler erstellt unter Beachtung von Timing-Vorgaben den korrekten Code für die State-Machine.

## Mikrocode des OP-Codes „ADC“

```
/* 82 - ADC abs */
{ 0x82,
  FETCH,
  NOP,
  MOVE_ROMPC_TO_ADDR_LO |
  INC_PC,
  MOVE_ACCU_TO_AREG1,
  MOVE_ROMPC_TO_ADDR_HI,
  INC_PC,
  MOVE_RAM_TO_AREG2,
  ALU_TO( MODE_ADC,
          CFLAG_VISIBLE,
          ALU_TO_ACCU ) |
  LOAD_FLAGS_ALU |
  NEXTOP,
},
```

# Assembler-Sprache

- Die Assembler-Syntax der MyCPU ist dem Prozessor 6502 sehr ähnlich.
- Es werden die meisten der 6502-Befehle unterstützt.
- Keine Binärkompatibilität !
- Es gibt keine relativen Sprünge, nur absolute (auch bedingte Sprünge sind bei der MyCPU immer absolut, da Zeitersparnis)
- Zusätzliche Befehle:
  - MUL / DIV für 8/16 Bit Integer Zahlen
  - 16 Bit Zeiger-Operationen: LPT, SPT, LPA, SPA
  - Befehle zur schnellen Berechnung von CRC-Summen
  - Befehle zur schnellen Umschaltung von Speicherseiten
  - erweiterte Befehle für Stack-Manipulationen
  - u.v.m.

# Der MyCPU - Emulator

## Definition:

Ein Emulator versucht eine reale Hardware so naturgetreu wie möglich auf einem anderen Computer nachzuahmen.

## Nutzen:

- Testen von Hardware, die noch nicht real existiert. Die CPU und die Grafikkarte wurden zuerst in Software emuliert, um deren Funktion sicherzustellen.
- Entwicklung von Software. Programme können auf dem PC geschrieben und im Emulator getestet werden. Der Emulator erlaubt vielfältige Debugging-Möglichkeiten, die die reale Hardware nicht bieten kann.

Den MyCPU-Emulator gibt es für MS Windows und BSD/Linux.

```
MyCPU Emulator Version 5
Press [R] to run, [Q] to quit, [S] to step over, other keys for singlestep.

PrgCntr  --CPU-Internal-Registers--  ---Flags---  ----- Page Registers -----
PC=0000  ACCU=00  X=00  Y=00  SP=00  C=0 Z=0 V=0  RAMP=00 ROMP=00 ZP=00 SP=01 D
0000  10A456      RESET      JMP  _Initcode

PC=56a4  ACCU=00  X=00  Y=00  SP=00  C=0 Z=0 V=0  RAMP=00 ROMP=00 ZP=00 SP=01 D
56A4  02          _Initcode  CLI

PC=56a5  ACCU=00  X=00  Y=00  SP=00  C=0 Z=0 V=0  RAMP=00 ROMP=00 ZP=00 SP=01 D
56A5  50FF      LDX #0xFF

PC=56a7  ACCU=00  X=ff  Y=00  SP=00  C=0 Z=0 V=0  RAMP=00 ROMP=00 ZP=00 SP=01 D
56A7  27          TXS

PC=56a8  ACCU=00  X=ff  Y=00  SP=00  C=0 Z=0 V=0  RAMP=00 ROMP=00 ZP=00 SP=01 D
56A8  2C          CLA

PC=56a9  ACCU=00  X=ff  Y=00  SP=00  C=0 Z=0 V=0  RAMP=00 ROMP=00 ZP=00 SP=01 D
56A9  42003E     STAA REG_RESERVED

PC=56ac  ACCU=00  X=ff  Y=00  SP=00  C=0 Z=0 V=0  RAMP=00 ROMP=00 ZP=00 SP=01 D
56AC  42003F     STAA REG_RESERVED
```

Debug- und  
Status-Fenster

Bildschirmausgabe  
der MyCPU

```
MyCPU CRT
FOUND: Memory Extension Board
FOUND: Interrupt Controller
FOUND: RS232 Interface (UART)
FOUND: Parallel Printer Port
FOUND: AT Keyboard Controller
FOUND: HD44780 LCD Display
FOUND: VGA Graphics Unit
FOUND: IDE Interface Controller

Installing IDE Controller:
Primary Master : EMU/DISK hda.bin / 2047MB
Primary Slave  : not present
Secondary Master : EMU/CDROM dummy
Secondary Slave  : not present

Mounting IDE drives:
Drive 8: Partition 0 on IDE Primary Master, size 2048KB
Drive 9: Partition 1 on IDE Primary Master, size 5120KB
Drive 10: Partition 2 on IDE Primary Master, size 5120KB
Drive 11: Partition 3 on IDE Primary Master, size 5120KB
4 hidden partitions not mounted

starting 8:/init

Please wait, initializing...
```



# Software für die MyCPU

Haupt-Programmiersprache: Assembler

- Es gibt 247 verschiedene OP-Codes. 9 OP-Codes sind noch undefiniert und können vom Anwender frei definiert werden (z.B. für Lehrzwecke).
- Ein OP-Code benötigt im Schnitt 9 Takt-Zyklen. Die MyCPU erreicht somit bei einer Taktrate von 8 MHz eine Geschwindigkeit von 0,89 MIPS.
- Es gibt zwei verschiedene Macro-Cross-Assembler für die MyCPU: ca65 (Unix/DOS) und hasm (nur DOS). Zusätzlich gibt es einen Assembler, der auch direkt auf der MyCPU läuft.

# Zusätzliche Programmiersprachen

## Basic

- Die MyCPU lässt sich in einem alten Basic-Dialekt, dem Basic V2 (bekannt vom Commodore 64), programmieren.
- Basic-Programme, die für den C64 geschrieben wurden, funktionieren auch auf der MyCPU.

## ANSI-C

- Es gibt einen C-Cross-Compiler (cc65, DOS/Unix) für die MyCPU. Die gängigsten C-Funktionen werden direkt unterstützt: Konsolen- Ein-/Ausgabe, Dateizugriffe, Standard-C Laufzeitbibliothek (Speicherverwaltung, String-Funktionen, Systemaufrufe).

# MyCPU Betriebssystem

- Das Betriebssystem ist zu 100% in Assembler geschrieben
- Das Betriebssystem belegt 160 kb ROM
- Bedient wird es über eine DOS-artige Eingabeaufforderung
- Alle gängigen DOS-Befehle sind vorhanden
- Bis zu 8 Speicherlaufwerke (RAM-Disk, Festplatte, CDROM, CF-Karte, Netzwerklaufwerk) werden unterstützt.
- Direkte Unterstützung von zwei seriellen (COM1: / COM2:) und einer parallelen (LPT1:) Schnittstelle
- Ein Basic-Interpreter ist integriert.

# Vorhandene Software

Folgende Software läuft auf der MyCPU:

- Basic-Interpreter für den schnellen Einstieg
- Text-Editor
- Assembler
- Netzwerk TCP/IP-Stack
- Telnet-Demon und HTTP Webserver
- Pico]OS - Echtzeit Betriebssystem
- Verschiedene Spiele
- Tools: Datei-Packer, Grafik-Viewer, und vieles mehr

# Datenaustausch mit der MyCPU

Oft ist es nötig, Dateien zwischen dem MyCPU-System und einem PC auszutauschen. Diese Aufgabe übernimmt der „Remote Filesystem“-Treiber:

## RemoteFS

- Die MyCPU wird über eine serielle Schnittstelle (RS232) mit einem Dateiserver verbunden.
- Ein beliebiges Verzeichnis auf dem Server-PC wird in der MyCPU als physikalisches Laufwerk abgebildet.
- Die MyCPU erhält Vollzugriff auf das freigegebene Verzeichnis. Alle DOS-Operationen sind möglich.  
Beispiel: **8:\>deltree 8:\windows**

# Nachbau

- Das gesamte MyCPU-Projekt ist „Open Source“.
- Auf [www.mycpu.thtec.org](http://www.mycpu.thtec.org) gibt es ein Forum zum Austausch von Ideen und zum Stellen von Fragen.
- Platinenlayouts im PDF-Format stehen zum Selberätzen der Platinen bereit. Fertige Platinen können erworben werden.
- Die gesamte Hardware ist Open Source, die Schaltpläne und Platinenlayouts können von der Homepage heruntergeladen werden (Eagle-Format)
- Software und alles Weitere, das zum Nachbau benötigt wird, steht auf [www.mycpu.eu](http://www.mycpu.eu) zum download bereit.

ENDE