

MICROCOMPUTER VO - Fragenkatalog

1. Wann muss eine Pipeline geflusht werden (2 Beispiele)?

- bei einem JMP Befehl, deren Ziel nicht bekannt ist
- Interrupt, ist naturgemäß unvorhersehbar
- Verzweigungsbefehle deren Verzweigungsargument nicht bekannt war.
EQN ($Req0 = Req0 + Req1$)

2. 4 Aufgaben der MMU?

- Adressberechnungen
- Verwaltung virtueller Speicher
- Schutzkonzepte
- Zugriff auf Arbeitsspeicher und sonstige Hardware
- Übersetzung der logischen Adresse in die physikalische Adresse

3. Charakteristik der „Von Neumann“ Architektur

- Rechner besteht aus CPU, Speicher Ein/Ausgabe
- Programmcode und Daten stehen im selben Speicher
- Aufbau eines Rechners ist unabhängig vom zu bearbeitenden Problem
- Speicher in gleichgroße Blöcke mit fortlaufender Nummerierung aufgeteilt
- Aufeinander folgende Befehle befinden sich in aufeinander folgenden Speicherzellen
- Im Steuerwerk gibt es einen Befehlszeiger der auf den nächsten Befehl zeigt

4. Was ist das Minimalitätsprinzip in der Mikrocomputerarchitektur?

- Von Neumann Architektur
- Nur die notwendigste Hardware implementiert
- Es kann nichts weggelassen werden
- Komplexität wird in die Zeit verlagert

5. Unterschied zwischen μC und μP ?

- μC hat alles „on Board“ und besteht aus einem μP , Speicher, und zusätzlicher Peripherie (I/O Controller, AD/DA Wandler, Timer, PWM Units, Watchdog, ...).
- μP besteht aus Steuerwerk, Rechenwerk, Schnittstelle zum Systembus, on – chip Cachespeicher und der Speicherverwaltungseinheit.

10. Was ist ein Refresh Zyklus und wozu dient er, wie läuft er ab?

Beim DRAM müssen regelmäßig die Kapazitäten nachgeladen werden, weil durch Leckströme die Information verschwinden. Dabei wird jede Speicherzelle gelesen und gleich darauf wieder mit dem selben Wert (also logisch 0 oder 1) beschrieben.

11. Warum ist 500MHz DSP bei graphischen Aufgaben schneller als ein 2GHz P4?

- Weil der DSP speziell auf Funktionen wie Multiplizieren, Datenverschieben ausgelegt ist
- Klassische Harvard Architektur (Daten und Befehlsrechner jeweils eigenständiger Speicher + Bussystem)
- DSP können durch spezielle Hardwareeinheiten arithmetische Funktionen in sehr kurzer Zeit bearbeiten (Multiplikation + anschließender Addition macht MAC Einheit)
- Bei DSP wird massiv auf Parallelisierung gesetzt, die durch spezielle Algorithmen effizient ausgenutzt wird.

12. Hauptvorteil eines Intel 80x86 gegenüber dem Motorola 68060?

- Intel 80x86 ist nach außen hin ein CISC Prozessor und intern werden die Befehle in RISC ähnliche Befehle zerlegt und im RISC Kern abgearbeitet
- Motorola ist ein ganz normaler CISC Prozessor (CISC Prozessoren haben typischerweise ein µProgramm und benötigen viele Taktzyklen zum Abarbeiten eines Befehls.
- Intel setzt auf Kompatibilität der Prozessoren (DOS läuft auch noch auf aktuellen Prozessoren), Motorola nicht

13. Zeichne eine 3 Stage Pipeline mit Fetch, Decode, Execute, Writeback (bis F5).

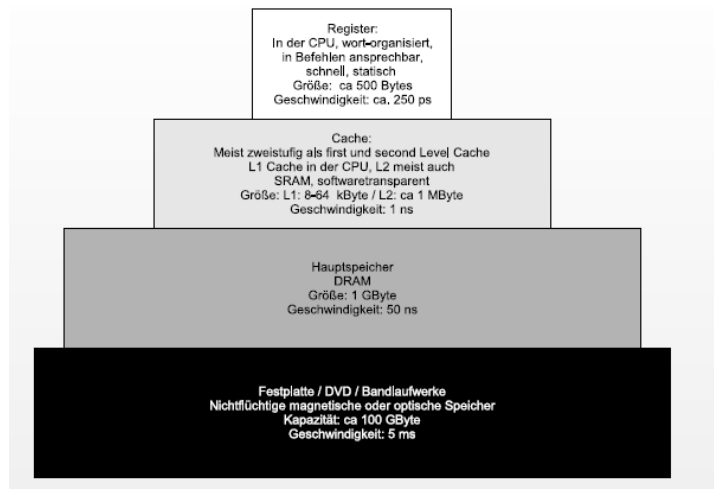
	F1	D1	E1	W1	F4	D4	E4	W4		
		F2	D2	E2	W2	F5	D5	E5	W5	
			F3	D3	E3	W3				

F...Fetch D...Decode E...Execute W...Write Back

14. Wie kann zwischen Speicher und E/A Controller Daten ausgetauscht werden?

- Über DMA (direct memory access) oder über den Prozessor

15. Speicherhierarchie: Zeichne und erkläre Wozu?



- Programmierer wollen „unendlich viel Speicher“. Um diesen Ziel in ökonomischer Weise nachzukommen, verwendet man eine möglichst ausgeklügelte Speicherhierarchie die sich auf dem Prinzip der zeitlichen und örtlichen Lokalität stützt. (großer billiger langsamer Speicher unten, teurer kleiner schneller Speicher oben)

16. Welche hierarchischen Ebenen gibt es beim Speicher eines μC und warum?

- Register (sehr schneller Speicher)
- interner Arbeitsspeicher (RAM) zur Programmausführung (meist als SRAM realisiert)
- interner Festspeicher in welchem das Arbeitsprogramm (Firmware) residiert (meist als ROM oder FLASH - Speicher realisiert)
- interner nichtflüchtiger Datenspeicher zur Ablage von Konfigurationen oder aktuellen Prozessparametern (meistens als EEPROM mit einigen Bytes bis einigen Dutzend kByte realisiert)

17. Welche Implementierung von Caches gibt es und welche hat den geringsten Hardwareaufwand?

- Voll assoziativer Cache
- Set assoziativer Cache

beim set assoziativen Cache ist der Hardwareaufwand am geringsten, weil beim voll assoziativen Cache ein CAM benötigt wird, der sehr komplex ist und vom Prinzip her schneller sein muss als das Cache - Data - Memory um einen Geschwindigkeitsvorteil auszunutzen zu können, dass ist aber in der Realität kaum zu realisieren, da das Cache – Data - Memory ohnehin schon ein sehr schneller Speicher ist.

18. Vergleiche Vektor- und Superskalarprinzip:

- Superskalarprinzip kann mehrere Befehle gleichzeitig ausführen, das erfolgt 100 % parallel weil alle nötigen Ressourcen mehrfach ausgeführt sind (MIMD). Der Prozessor entscheidet während dem Betrieb was parallel gemacht werden kann und was seriell gemacht werden muss.
→ Hardware komplexer
- Vektorrechner implementieren lediglich SIMD (mehrere Rechenwerke + Speicher aber nur ein Steuerwerk) der entschlüsselte Befehl kann aber gleichzeitig auf mehrere Operanden in mehreren Rechenwerken angewandt werden.

19. Wie viele Zeiger braucht FIFO und LIFO?

- FIFO benötigt zwei Zeiger
- LIFO: 1 Zeiger

20. Wie funktioniert eine μ Programmsteuerung?

- softwaremäßige Implementierung des Steuerwerks:
- Dabei wird das jeweilige Mikroprogramm der Makrobefehl (eigentliche Maschinenbefehle) in den Mikroprogrammspeicher geladen und sequentiell abgearbeitet

21. Nenne 4 Charakteristika von RISC?

- Wenig, dafür schnelle Befehle
- 1 Befehl pro Takt
- Häufigsten Befehle implementiert
- Nur LOAD/STORE Befehle greifen auf Speicher zu
- Verwendet Pipelining
- Steuerwerk in Hardware

22. Welche Daten gibt es neben den Nutzdaten beim Cache?

- TAGS: Adresse im Hauptspeicher
- FLAGS: Dirty, Used, ect.
- Frame: Kopie der Hauptspeichereinhalte - Daten

23. Beschreibe Write Through.

- Der Inhalt im Hauptspeicher und im Cache müssen jederzeit auf aktuellem Stand sein. Jede Schreiboperation wird zusätzlich noch zum Hauptspeicher weitergereicht (durchgeschrieben).
→ Hauptspeicher viel langsamer als Cache, es kommt zu Verzögerungen
→ Abhilfe: Pufferspeicher

24. Was ist die „semantische Lücke“, wie hängt sie mit der von Neumann Architektur zusammen und welche beiden Aspekte hat diese Lücke?

- Ein Von Neumann Rechner ist nicht in der Lage ein Objekt zu erkennen und zerlegt es in eine Bitkette (ein Java Objekt hat keine Unterstützung vom Prozessor und wird in eine Bitkette aufgespaltet mit der Neumann Rechner umgehen kann).
- Grund: Daten und Befehle liegen hintereinander im Speicher, es kann nicht zwischen Befehlen und Daten unterschieden werden → erkennt kein Objekt.
- Uneingeschränkte Zugriffsmöglichkeit auf jedes beliebige Speicherobjekt (verhindert MMU)
- 2 Aspekte:
 1. Daten und Befehle können nicht unterschieden werden, beide sind Bitketten im gleichen Speicher, deshalb uneingeschränkter Speicherzugriff
 2. von Neumann Rechner ist nicht in der Lage, die in höheren Programmiersprachen verwendeten Typen-Attribute für Datenobjekte zu erkennen bzw. zu verarbeiten (semantischer Zwischenraum zwischen Maschinen- und Hochsprache)

25. Vorteil der Thumb-Erweiterung des ARM?

- Thumb Befehle halb so lang wie normale ARM befehle
- Ausführung in 32 Bit Kernen mittels Dekompressor (wandelt 16Bit Thumbcode in 32 Bit ARM code ohne Performanceverlust um)
- Thumb ist Submenge des ARM Befehlessatz
- 30 - 40% kürzerer Code
- Thumb und ARM Code ist mischbar → bei umschalten wird Pipeline geleert
- Weniger Speicher → Einsatz für mobile Geräte

26. Welcher Prozessor ist für Server besser geeignet, einer mit feiner oder grober Parallelisierung?

- Feiner Parallelisierung wird versucht Befehle einer Befehlssequenz zu finden die Parallel abgearbeitet werden können (die nicht voneinander abhängen)
- Grober Parallelisierung (Threads, Prozesse, Multi Clients sind wirklich parallel)
- Bei Serveranwendungen ist hauptsächlich load - balancing und Latenzzeit wichtig, und die Anfragen der Clients sind meist dieselben, deshalb ist grobe Parallelisierung (mehrere komplette Rechnersysteme) besser geeignet.

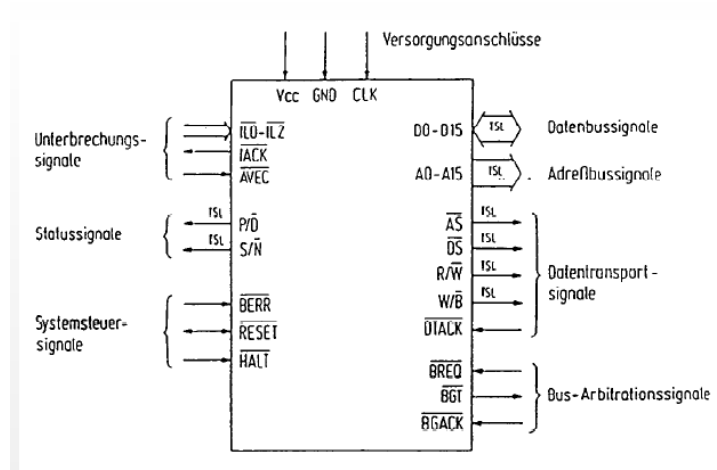
27. Lösungen für Pipeline – Ressourcenkonflikte

- Zwei Pipelinestufen benötigen dieselbe Ressource (den Bus, Multiplizierer ect.)
- Interlocking (kurze Pause)
- Schnellere Ressource
- Ressourcenduplizierung

28. Ein Big Endian Rechner speichert 0xABCD1234, Was liest ein Little Endian Rechner an dieser Stelle 0x.... ?

- 0x3412CDAB

29. Vier typische externe Steuerepins eines μ -P, plus ihre Bedeutung und Verwendung? (außer HALT, RESET, CLK):



- AS Address Stable
- DS Data Stable
- AVEC Auto Vector
- IACK Interrupt Acknowledge
- BERR Bus Error
- IL Interruptlevel
- BGACK Bus Grant Acknowledge
- BGT Bus Grant
- BREQ Bus Request
- DTACK Data Acknowledge

30. 4 Architekturelle / technologische Methoden, um die Verlustleistung von μ Chips zu verringern?

- Bessere Isolierungen \rightarrow Tunnelströme verringern
- Serienschaltungen von Transistoren reduzieren den Leckstrom um den Faktor 5-10
- Multithread Prozessoren und intelligentes Caching \rightarrow keine unproduktiven Wartezeiten
- Geringere Betriebsspannungen

31. Phänomene, die zu dynamischen Verlustleistungen bei CMOS führen?

- Schaltvorgänge bei denen unvermeidbar Kapazitäten umgeladen werden
- Wenn beide Schalten → kurzzeitiger Kurzschluss → Kurzschlussstrom

32. Was ist neben der Performance des Computersystems in der Maßzahl eines Benchmarktests noch enthalten?

- Nicht nur die Hardware des System selber wird getestet sondern auch Compiler und die verwendete Software

33. Wozu dient der Translation Lookaside Buffer, was ist er und was leistet er?

- Er ist eine Funktionseinheit der MMU (Memory Management Unit)
- Beim Umrechnen von virtuellen Speicher in physikalischen Speicher muss mit Übersetzungstabellen gearbeitet werden. → sehr zeitintensiv und deshalb wird im TLB die Page Tabelle gepuffert.
- TLB kann begrenzte Mengen dieser Daten halten (32 bis 64 Einträge) und beschleunigt den Vorgang damit.
- Erlaubt durch assoziative Ordnungsregister Parallele zugriffe und ist dadurch sehr schnell aber auch teuer.

34. Unterschied zwischen Pages und Segments?

- Pages: Unterteilung des Speichers in gleich große Speicherblöcke
- Segmente: Unterteilung des Speichers in Speicherblöcke verschiedener Größe
- (Problem: Fragmentierung)
- Abhilfe: Segementiertes Paging

35. Wann werden bei DDRSDRAM Daten übergeben?

- Bei steigender und fallender Flanke

36. Welches Problem löst Interleaving bei DRAM?

- Wenn von zwei nacheinander Folgenden Adressen gelesen wird, muss nicht gewartet werden bis der Refreshzyklus abgeschlossen ist sondern es wird von einer anderen Speicherbank gelesen während bei der anderen noch geschrieben wird.

37. Programmspeichergröße und Taktrate einer typischen Smart Card?

- 66Mhz
- 32 Bit
- 1Mbyte EEPROM
- 400kByte RAM

38. Schritte der IC-Produktion chronologisch sortieren (dotieren, belichten, ätzen, entwickeln)?

- Belichten, Entwickeln, Ätzen, Dotieren

39. Was sind Benchmarks?

- Sind Programme die allgemeine oder spezielle Anforderungsprofile repräsentieren, Ziel ist es Leistungsfähigkeit verschiedener Systeme möglichst objektiv vergleichen zu können.

40. Was ist ein Pikobefehl

- Bei horizontalen Mikroprogrammen ist das Mikro Codewort sehr breit und besteht aus einzelnen Bits von denen jedes direkt einer Steuerleitung des Steuerwerks entspricht.
- Diese Bits werden auch als Pikobefehl bezeichnet.

41. Unterschied zwischen static und dynamic scheduling beim Auflösen von Pipeline Steuerkonflikten.

- Static Scheduling: Befehle werden vom Compiler umsortiert
- Dynamic Scheduling: Der Prozessor lädt mehrere Befehle und sortiert diese für eine effizientere Abarbeitung um.

42. 3 stufige Pipeline mit 4 Zyklen (FE, DE, EX, WB) -> wann DE 5?

- In der siebten Phase

43. Was ist Virtueller Speicher

- Wenn der Arbeitsspeicher nicht ausreichend für meine Anwendung ist, so müssen Daten aus dem Arbeitsspeicher z.B. auf die Festplatte ausgelagert werden.
- Problem ist dabei, dass ein größerer logischer Adressbereich auf einen kleineren Physikalischen Adressbereich abgebildet werden muss.
- Für den Virtuellen Speicher wird meistens eine MMU verwendet. Dort werden die logischen Adressen in Real Adressen mit Hilfe von Tabellen umgesetzt die vom Betriebssystem verwaltet werden.

44. Was ist SIMD?

- Rechnerklassifikation nach Flynn 4 Klassen: Singel Instruktion Stream - Multiple Data Stream (z.B.: ein Steuerwerk, mehrere ALU + Datenbus + Datenspeicher)

45. Was bezeichnet man als Firmware?

- Die Software auf dem Speicher eines μC
- Enthält den Programmcode

46. Was versteht man unter Datenabhängigkeit bei Pipeline Konflikten?

- Ein Befehl benötigt das Ergebnis eines anderen Befehls, dessen Abarbeitung noch nicht abgeschlossen ist

47. Was versteht man unter Befehlsabhängigkeit bei Pipeline Konflikten?

- Steuerungsflusskonflikt
- z.B.: jumpif Reg3, DestAddr (wenn Reg3 noch nicht beschrieben mit neuem Wert → Falsche Sprungbedingung) → Einführung von waitstates oder Umsortierung

48. Ein Vorteil und ein Nachteil einer μ Programmsteuerung

- Hard-wired ist schneller als Mikroprogrammsteuerung
- VT Mikroprogramm ist übersichtlicher
- VT Mikroprogramm hat leichte Änderbarkeit um Fehler auszubessern
- Hard-wired kann Hardware sehr komplex werden, deshalb wird die Anzahl der unterstützten Befehle gering gehalten
- NT Mikroprogramm geringe Geschwindigkeit
- VT Mikroprogramm ist Flexibler und bietet mehr Möglichkeiten

49. Jeweils zwei Beispiele für SIMD & MIMD -> nur jeweilige Architektur angeben

- SIMD → VLIW, Vektorrechner, Superskalarrechner
- MIMD → multicore on chip, multithreaded Prozessoren, multichip

50. Unterschied zwischen Write - Through und Copy - Back

- Copy Back speichert bei vollem Cache die Daten in den Arbeitsspeicher
- Write Through leitet die Daten direkt weiter an den Arbeitsspeicher
- Copy Back ist schneller, NT: Daten sind währenddessen inkonsistent

51. Unterschied Mikrobefehl, Makrobefehl & Beziehung der beiden zueinander

- Makrobefehle sind die eigentlichen Maschinenbefehle (Command Set des Prozessors)
- Mikrobefehle sind die kleinsten Kommandos (Makrobefehl besteht aus vielen Mikrobefehlen)
- Wenn ein Makrobefehl ausgeführt werden soll, wird das zugehörige Mikroprogramm im Mikroprogrammspeicher gesucht und ausgeführt
- Minimalitätsprinzip (1 Befehl mehrmals für eine Operation ausführen)

52. Vorteile von modularen Design (4 Stück)

- Wiederverwendbarkeit, Austauschbarkeit
- Klarheit und Übersichtlichkeit (getrennte Betrachtung)
- Module in Teams getrennt übergeben (Verkürzung Entwicklungszeit)
- Testen der einzelnen Teile
- Controlling (kosten Explosion)

53. Bei welchen Speicherarten ist Interleaving möglich?

- DRAM
- Harddisk
- Optische Laufwerke (CDs, DVDs, ..)

54. Kann bei Verwendung eines DSP's in einem öffentlichen Netz eine "Buffer - Overflow - Attacke" durchgeführt werden? Ja/Nein. Begründen Sie.

- Kann es nicht weil Daten und Code im getrennten Speicher stehen (Harvard Architektur!!)

55. Welche Aufgabe kommt dem Compiler bei der VLIW - Architektur zu?

- Der Compiler fasst mehrere sequentielle Programmbefehle zu einem langen Superbefehlswort zusammen, das in einem Taktzyklus ausgeführt wird.
- Das vorausblickende und schlaue anordnen der Befehle übernimmt der Compiler
- Prozessor bekommt Befehle bereits in wunderbar parallel ausführbaren Paketen geliefert.
- Compiler versucht das Programm (die Maschinenbefehle) so effizient wie möglich zu parallelisieren (auf die div. Exe-Einheiten zu verteilen)

56. 4 Vorteile eines IC's gegenüber einer diskreten Transistor – Schaltung

- Kleiner
- Billiger
- Höhere Schaltgeschwindigkeit
- Sparsamer beim Stromverbrauch

57. Kriterien, die ein universeller digitaler Signalprozessor erfüllen sollte!

- Effiziente Arithmetik (z.B. Multiplizieren + Speichern)
- Effiziente Schleifen für Signalflüsse
- Mehrere Busse
- Spezielle Betriebsmittel (digitale Filter, ect...)
- Harvard Architektur
- Breite Busse, gute interne Caches
- Eventuelle 2 Datenspeicher für Zufuhr von 2 Datenströmen

58. 4 Vorteile & Nachteile von Gleitkommadarstellung!

- + größere Zahlendynamik und Präzision
- + besser für den Einsatz von C-Compilern
- + Genauigkeit konstant
- + kaum mehr Skalierungsoperationen
- nicht jede beliebige Bitkombination ist eine gültige Zahl
- Variablen müssen immer initialisiert werden
- kein einheitliches Zahlenformat
- höherer Hardwareaufwand im Gegensatz zur Festkommadarstellung

59. Vorteile & Nachteile von Festkommazahlendarstellung?

- Einfacher zu Implementieren
- Nur kleine Zahl 2^{Bitzahl}

60. Vorteile & Nachteile von Pipelining!

- Viele Abläufe können unterteilt werden → schneller
- Ausnützung aller Ressourcen
- Achten auf Datenkonsistenz (Data, Ressource, Controllhazards)
- Nops müssen eingeführt werden → es kann zu einer nicht effektiven Nützung führen

61. Unterschiede zwischen Superskalar und VLIW?

- Bei Superskalare Prozessoren übernimmt der Prozessor selbst das Umsortieren, Anordnen und Optimieren der Befehlskette. Bei superskalaren DSP's erfolgt Parallelverarbeitung mit vorgegebenen Mehrfacharithmetikbefehlen.
- Superskalar: MIMD
- Bei VLIW übernimmt diese Aufgabe der Compiler
- VLIW: SIMD

62. 4 Adressierungsarten für DSP's!

- Register – indirekte Adressierung (mit nachträglicher Inkrementierung)
- Register – direkte Adressierung
- Modulo – Adressierung
- Bit – Reverse Adressierung (für FFT)

63. Welche Funktionseinheiten sollte ein moderner integrierter Signalprozessor besitzen?

- jeder Befehl soll nur einen Taktzyklus dauern
- Befehlssatz soll gut an DSP - Aufgaben angepasst sein
- MAC mit großer Genauigkeit
- Adress- und Datenverarbeitungsoperationen parallel
- getrennte Speicher und Bussysteme zumindest für Befehle und Daten
- Register ohne Einschränkung bei jedem Befehl verwendbar
- Bei Schleifen -> Schleifenzähler und Stack automatisch verwaltet
- Bei Indirekter Adressierung -> bit reversed und Ringpuffer möglich
- SIMD- und (eingeschränkte) MIMD-Befehle sollen vorgesehen sein

64. Welche Eigenschaften hat die Zweierkomplementdarstellung?

- Msb: Vorzeichen (1: neg. 0: pos.)
- Das Vorzeichen einer Zahl wird geändert, in dem alle n bit bitweise invertiert werden und anschließend 1 addiert wird
- Nur eine Darstellung für die Zahl 0
- Wertebereich mit n-bits: $-2^{(n-1)}$ - $(2^{(n-1)} - 1)$...pos. Eine Zahl weniger

65. Welche Aufgabe hat ein Cache bei Signalprozessoren

- Zwischenspeicher der Programmbefehle
- Speziell bei Schleifen, Programmspeicher entlasten

66. 4 CPU Architekturen nach dem Registerzugriff (Akku Maschine, Adress..)

- Stack Maschine
- Accu Maschine
- Register Memory Maschine
- Register Maschine (LOAD/STORE Architektur)

67. Unterschied zwischen HW-Software Monitor

- HW: Sind elektrische Messinstrumente, die an internen Kontaktpunkten Zustände des Systems untersuchen.
- Software: Spezielle Messprogramme die in die Software des zu beobachteten Systems eingebettet sind.

68. 4 Typische ALU Flags

- (N) → Negativ Flag
- (V) → Overflow Flag
- (Z) → Zero Flag
- (C) → Carry Flag

69. Was ist der Unterschied zwischen Kernprogrammen und synthetischen Programmen bei der Leistungsbeurteilung?

Synthetische Programme sind eine willkürliche Mischung von Operationen die nicht unbedingt in funktionalem Zusammenhang stehen. Sie werden nur zum Zweck der Rechnerbewertung erstellt und zum Laufen gebracht. Synthetische Programme bieten aber auch einen Vorteil. Da sie keine produktiven Ergebnisse liefern müssen, eignen sie sich gut zur Erzeugung von Lasten beliebiger Ausprägung und damit zum Beleuchten ganz bestimmter Eigenschaften der getesteten Maschine.

Kernprogramme sind typische Anwenderprogramme die für den zu bewertenden Rechner geschrieben werden. Sie werden jedoch nicht zur Ausführung gebracht. Ziel ist es, die Ausführungszeit auf Grundlage der einzelnen Befehlsausführungszeiten zu bestimmen. Das ist zum einen sehr aufwändig, zudem ist es nicht möglich, Parallelitäten in der Ausführung zu berücksichtigen. Darum sind Kernprogramme aber auch Mixe bei modernen Prozessorarchitekturen nur bedingt sinnvoll anwendbar.

70. Welche 2 Arten der Vektoradressierung gibt es?

- Tabellarische Zuordnung zw. den logischen und den physikalischen Adressen (indirekte Adressierung)
- Abbildung nach linearer Funktion $y = k \cdot x + d$
 - y... physikalische Adresse x
 - d... Teile der logischen Adresse
 - k... Parameter (meist eine 2er Potenz)

71. Was versteht man unter SIMD-Erweiterungen (& Anwendungsbeispiele)?

Eine sehr verbreitete Erweiterung der Rechenfähigkeit einer CPU sind Betriebsmittel für SIMD Befehle. Besonders Multimedia-Anwendungen (und hier im speziellen Video - Rendering) hat es erforderlich gemacht, dass "normale" Universal - CPUs und Desktop - CPUs spezialisierte Betriebsmittel bekommen:

- MMX (Multi-Media Extension, 64-Bit Integer SIMD, 57 neue Befehle)
- 3DNow! (von AMD, 32-Bit Integer/Float SIMD a.k.a. single precision, 21 neue Befehle)
- SSE (Streaming SIMD Extensions im Katmai Kern von Intel [Pentium III], ebenso 32-Bit Integer/Float SIMD, 70 neue Befehle)
- SSE2 (64-Bit double precision float SIMD)
- SSE3 (im Intel Prescott Core sowie im AMD Athlon64, 13 neue Befehle im Vergleich zu SSE2)
- AltiVec (die Integer/Float SIMD Erweiterung im Power Chip, so genannt von Motorola. Um die Verwirrung zu steigern, nennt IBM diese Technologie VMX, Apple hingegen nennt sie Velocity Engine)

72. Was ist horizontale Mikroprogrammierung bzw. was ist vertikale Mikroprogrammierung?

Mikroprogramme werden klassischerweise in der CISC Architektur verwendet, fest verdrahtete Steuerwerke in RISC.

Sehr breiter Mikrocode (auch als „horizontale Verlagerung“ bezeichnet) sieht aus wie VLIW, sehr schmaler

(auch als „vertikale Verlagerung“ bezeichnet) sieht eher aus wie RISC.

Bei horizontalen Mikroprogrammen ist das Mikro-Codewort sehr breit und besteht aus einzelnen Bits, von denen jedes einzelne direkt einer Steuerleitung des Steuerwerks entspricht. Diese einzelnen Bits werden auch als Pikoprogramm - Befehle oder kurz Pikobefehle bezeichnet.

Bei vertikalen Mikroprogrammen beinhaltet jedes Mikroprogrammwort lediglich einen Pikobefehl, ein Makroprogrammwort besteht in diesem Fall also aus einer mitunter sehr langen

Mikroprogrammwortkette. Der Aufbau ist einfach, die Worte sind schmal, aber die Abarbeitung dauert länger.

73. Wie funktioniert assoziativer Speicher?

Ein Assoziativrechner verwendet, wie der Name schon suggeriert, einen Assoziativspeicher (content addressable memory, CAM, ein Konzept das an sich Zuse schon 1943 vorgeschlagen hat). Diesen Speicher adressiert man nicht mit einer Adresse sondern mit Inhalten. Das Laden der Telefonnummer des Herrn Maier (abgespeichert in einer Tabelle im RAM) in ein Register R1 führt also nicht zu einer Schleife, die den Speicher nach dem String "Maier" absucht, sondern lediglich zu der Abfrage "READ R1, Maier". Es wird also Funktionalität (und daher Rechenlast) auf den Speicher (!) ausgelagert.

74. Daisy Chain

Eine Daisy-Chain ist ein klassisches Beispiel für eine Dezentrale Busarbitration, das heißt, die einzelnen Master erkennen selbstständig, ob sie den Bus benutzen dürfen.