

Mikrocomputer Fragenkatalog

etstudent123

2. Mai 2017

Disclaimer: Die Fragen wurden aus alten Prüfungsangaben abgeschrieben. An der Formulierung der Fragen wurde nichts geändert. Es könnte sein das sich ein oder zwei Tippfehler eingeschleust haben. Ich übernehme keine Haftung auf Richtigkeit der Fragen oder sonst was.

Fragen vom 26.1.2017

1. Nennen Sie vier Speicher in einer möglichen Memoryhierarchie und beschreiben Sie jeweils deren Eigenschaften. Quantifizieren Sie deren Zugriffszeiten im Verhältnis zum Prozessortaktzyklus. (5.Punkte)

2. Manche C-Compiler erlauben eine 'asm()' Direktive. Dient diese dazu um: (5. Punkte)

- ☒ Einen Assembler als separaten Compilerungsschritt aufzurufen,
- ☒ Ein Assembler Statement in das erzeugte Assemblerprogramm einzufügen,
- ☒ Assemblerspezifische Optimierungen zu ermöglichen, oder
- ☒ Breakpoints zu setzen?

3. Wenn ein Word aus vier Bytes (32 bit) besteht, was ist Word Alignment: (5. Punkte)

- ☒ Wenn nur entweder die ersten beiden oder die letzten beiden Bytes verwendet werden,
- ☐ wenn alle Bytes in einem Word verwendet werden,
- ☐ wenn für alle Wordadressen gilt : Adresse mod 4 = 0,
- ☐ wenn für alle Wordadressen gilt : Adresse mod 4 = 0 oder Adresse mod 4 = 2, oder
- ☐ wenn die Wordlänge gleich der Größe einer Cacheline ist und alle Wörter in genau einer Cacheline passen.

4. Beschreiben Sie zwei Methoden Parameter an eine Subroutine zu übergeben. (5.Punkte)

5. Was ist ein Vectored Interrupt= Beschreiben Sie dessen Funktion und die dazu notwendigen Hardware und Softwarekomponenten. (5.Punkte)

6. Beschreiben Sie zwei wichtige Funktionen eines Betriebssystems. (5.Punkte)

7. Beschreiben Sie genau, wie ein Breakpoint und wie er vom Debugger verwendet wird. (5.Punkte)

8. Beschreiben Sie das Register-Transfer Model im Schaltungsentwurf (5.Punkte)

9. Motivieren und beschreiben Sie das δ -Modell von VHDL. (5.Punkte)

10. Was ist das Entscheidungsintervall bei einem D-Flip-Flop? Was bedeutet die Verletzung des Entscheidungsintervalls und was kann das bewirken? (5.Punkte)

11. Skizzieren Sie einen hierarchischen Carry look-ahead Addierer mit zwei Ebenen von Carry look-ahead Logik der beiden Ebenen. (5.Punkte)

12. Skizzieren Sie die Struktur eines 4-2 Reducer basierenden Multiplizierers. Was ist sein Hauptvorteil gegenüber einem Array Multiplizierer. (5.Punkte)

13. Erklären Sie den Unterschied der 'copy-back'- und der 'write-through'-Technologie bei Caches. Worin liegen jeweils die Vor- und Nachteile? (5.Punkte)
14. Beschreiben Sie den Aufbau einer 32 bit IEEE Floating Point Zahl und erklären Sie die Bedeutung der einzelnen Teile. (5.Punkte)
15. Erklären Sie die Begriffe 'predict taken' sowie 'predict not taken' im Zusammenhang mit einer Pipeline. Was soll mit diesen Techniken grundsätzlich vermieden werden? (5.Punkte)
16. Worin liegen die Probleme, eine Pipeline mit sehr vielen Stufen zu verwenden ? (5.Punkte)
17. Beschreiben Sie je einen Vor- und Nachteil von 'Associative Mapping' im Vergleich 'Direct Mapping' bei Caches? (5.Punkte)
18. Welche Funktion hat die Memory Management Unit(MMU)? (5.Punkte)
19. Erklären sie die Vor- und Nachteile von DRAMs gegenüber SRAMs. (4. Punkte)
20. Was versteht man bei einer Pipeline unter dem Begriff 'forwarding'? Erläutern Sie das an einem Codebeispiel. (4.Punkte)

Fragen vom 16.3.2017 Gruppe B

1. Erklären Sie den Unterschied der 'copy-back'- und der 'write-through'-Technologie bei Caches. Worin liegen jeweils die Vor- und Nachteile? (5.Punkte)
2. Erklären Sie die Begriffe 'predict taken' sowie 'predict not taken' im Zusammenhang mit einer Pipeline. Was soll mit diesen Techniken grundsätzlich vermieden werden?(5.Punkte)
3. Worin liegen die Probleme, eine Pipeline mit sehr vielen Stufen zu verwenden ? (5.Punkte)
4. Beschreiben Sie je einen Vor- und Nachteil von 'Associative Mapping' im Vergleich 'Direct Mapping' bei Caches?
5. Welche Funktion hat die Memory Management Unit(MMU)? (5.Punkte)
6. Erklären Sie die Vor-und Nachteile von DRAMs gegenüber SRAMs.
7. Was versteht man bei einer Pipeline unter dem Begriff 'forwarding'? Erläutern Sie das an einem Codebeispiel. (5.Punkte)
8. Beschreiben Sie den Aufbau einer 32 bit IEEE Floating Point Zahl und erklären Sie die Bedeutung der einzelnen Teile. (5.Punkte)
9. Skizzieren und beschreiben Sie einen sequenziellen Dividierer. (5.Punkte)
10. Erklären Sie das Prinzip der Carry Save Addition. Was ist ihr Hauptvorteil? (5.Punkte)
11. Ein Interrupt Vector ist (mehr als eine Antwort kann richtig sein): (5.Punkte)
 - ⊗ Ein Code zum Identifizieren eines Interrupts,
 - ⊗ eine Liste von Argumenten für die Interrupt Service Routine,
 - ⊗ eine Adress-offset der zum Programm Counter Addiert wird um zur Interrupt Service Routine zu verzweigen, oder
 - ⊗ das Status Register des Interrupt Controllers ?
12. Beschreiben Sie genau, wie ein Breakpoint funktioniert und wie er vom Debugger verwendet wird. (5.Punkte)
13. Zu welchen Aufgaben werden Interrupts vom Operating System verwendet? Nennen und beschreiben drei Aufgaben und beschreiben Sie das Zusammenspiel von Interrupts und Operating System Aktivitäten. (5.Punkte)
14. Was ist eine Interrupt Service Routine? Beschreiben Sie deren Funktionen und aus welchen Teilen sie üblicherweise besteht. (5.Punkte)
15. Skizzieren Sie die Struktur eines 4-2 Reducer basierenden Multiplizierers. Was ist sein Hauptvorteil gegenüber einem Array Multiplizierer. (5.Punkte)
16. Nennen Sie vier Speicher in einer möglichen Memoryhierarchie und beschreiben Sie jeweils deren Eigenschaften. Quantifizieren Sie deren Zugriffszeiten im Verhältnis zum Prozessortaktzyklus.(5.Punkte)
17. Motivieren und beschreiben Sie das δ -Modell von VHDL. (5.Punkte)
18. Skizzieren und beschreiben Sie Funktion und Aufbau eines Mealy Zustandsautomaten (5.Punkte)

19. Skizzieren und beschreiben Sie Funktion und Aufbau eines RS-Latches. (5. Punkte)
20. Skizzieren und beschreiben Sie Funktionsweise und Aufbau einer Single Instruction Multiple Data (SIMD) Rechnerarchitektur. (5. Punkte)

Fragen + Antworten vom 16.3.2017 Gruppe B

1. Erklären Sie den Unterschied der 'copy-back'- und der 'write-through' - Technologie bei Caches. Worin liegen jeweils die Vor- und Nachteile? (5.Punkte)(4.5)

A: write-through: Cache und Speicher wird geupdated effizient bei einmaligen Speicherzugriff ineffizient bei mehrmaligen Zugriff auf die selbe Speicherstelle.
copy-back: Cache und dirty-bit wird geupdated, Speicher erst später. effizient bei mehrmaligen Speicherzugriff, ineffizient bei einmaligen Zugriff.

2. Erklären Sie die Begriffe 'predict taken' sowie 'predict not taken' im Zusammenhang mit einer Pipeline. Was soll mit diesen Techniken grundsätzlich vermieden werden? (5.Punkte)(4.5)

A: Es geht darum, Sprünge vorausszusehen.

predicted taken: Es wird angenommen, dass der Sprung/Branche genommen wird.

predicted not taken: Es wird angenommen, dass der Sprung/Branche nicht genommen wird.

Um zu vermeiden \Rightarrow 1-Cycle ?proally?

3. Worin liegen die Probleme, eine Pipeline mit sehr vielen Stufen zu verwenden? (5.Punkte) (5)

A: - Komplizierter Hardwareaufbau - Sprünge/Branches sind komplizierter durchführbar, weil Pipeline geflushed werden muss

- mehr Rechenaufwand

- nur sinnvoll wenn alle Prozesse exakt gleich lang dauern

- es zu langen Wartezeiten kommt.

4. Beschreiben Sie je einen Vor- und Nachteil von 'Associative Mapping' im Vergleich 'Direct Mapping' bei Caches? (5.Punkte) (5)

A: Direct Mapping :

Vorteil: wenig Vergleicher für Tags

Nachteil: belegte Adressen können überschrieben werden obwohl noch Platz im Cache ist.

Associative Mapping:

Vorteil: Platz im Cache wird voll genutzt

Nachteil: viel Hardwareaufwand durch viele Vergleicher für Tags?

5. Welche Funktion hat die Memory Management Unit(MMU)? (5.Punkte)(3.5)

A: - koordiniert die Speicher

- lädt Speicheradressen aus Page Table wenn nicht in MMU sind (Es hören zu viel um alle in MMU zu schreiben)

- verwaltet virtuellen Speicher und sendet die richtigen Daten zu den richtigen Stellen

6. Erklären Sie die Vor-und Nachteile von DRAMs gegenüber SRAMs.(5.Punkte)
(4)

Vorteil: billig, groß, weniger komplexe Hardware

Nachteil: flüchtig, müssen oft refreshed werden wegen Leckströmen, langsam, Verbrauch höher

7. Was versteht man bei einer Pipeline unter dem Begriff 'forwarding'? Erläutern Sie das an einem Codebeispiel. (5.Punkte) (5)

A: Die Lösung einer Operation wird nicht nur ins Register zurückgeschrieben sondern gleich wieder der ALU übergeben zum weiter rechnen.

Codebsp.:

add R3, R4

sub R5, R3

8. Beschreiben Sie den Aufbau einer 32 bit IEEE Floating Point Zahl und erklären Sie die Bedeutung der einzelnen Teile. (5.Punkte) (4 oder 3)

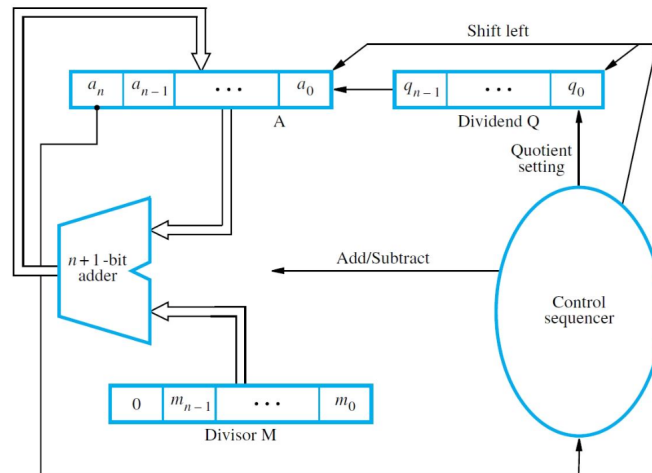
Sign gibt Vorzeichen an (1bit)

Exponent (8bit)

Mantisse (23bit) enthält die Zahl

die Vorkomma-Eins wird als Hidden Bit weggelassen; da dort immer eine 1 steht, braucht man diese nicht zu speichern

9. Skizzieren und beschreiben Sie einen sequenziellen Dividierer. (5.Punkte) (0)
 Ein sequenzieller Dividierer besteht aus 3 n-bit Register, einen n-bit Addier-/Subtrahierwerk und einem Kontrollsequenzer. Eine Sequenz aus n Addition/Subtraktion Zyklen generiert einen n-bit Quotienten und einen n-bit Rest. (Folieninhalt ...)



10. Erklären Sie das Prinzip der Carry Save Addition. Was ist ihr Hauptvorteil? (5.Punkte) (3)

A: Schneller, kann mit Vorzeichen rechnen, Speichert Carry von einer Stufe zur nächsten. Bei jeder Addition wird der Betrag und das Carry extra gespeichert und an die nächste Stufe übergeben. (3.Punkte)

11. Ein Interrupt Vector ist (mehr als eine Antwort kann richtig sein): (5.Punkte)

- ⊗ Ein Code zum Identifizieren eines Interrupts,
- ⊗ eine Liste von Argumenten für die Interrupt Service Routine,
- ⊗ eine Adress-offset der zum Programm Counter Addiert wird um zur Interrupt Service Routine zu verzweigen, oder
- ⊗ das Status Register des Interrupt Controllers ?

12. Beschreiben Sie genau, wie ein Breakpoint funktioniert und wie er vom Debugger verwendet wird. (5.Punkte) (5)

A: Das Programm wird an einer bestimmten Stelle angehalten und man kann step for step nachvollziehen, welchen Wert Variablen annehmen. Dazu wird eine ISR (Interrupt Service Routine) vom Debugger aufgerufen. Dabei darf das Programm nicht verändert werden.

13. Zu welchen Aufgaben werden Interrupts vom Operating System verwendet? Nennen und beschreiben drei Aufgaben und beschreiben Sie das Zusammenspiel von Interrupts und Operating System Aktivitäten. (5.Punkte) (5)

z.B. zum Ausführen von Unterprogramm (z.B. Rechenoperationen)

- zum Booten wird eine ISR und ein Interrupt benötigt
- falls eine dringende Aufgabe zuerst erledigt werden muss.
- wenn ein Programm lange dauert und die freie Zeit für ein anderes Programm genutzt werden soll

Interrupts werden aufgerufen → PC wird in Sicherheit gebracht → Interrupt wird ausgeführt und signalisiert OS das es fertig ist → PC wird geladen und Programm läuft an der richtigen Stelle weiter. (Allgemein: OS steuert alles)

14. Was ist eine Interrupt Service Routine? Beschreiben Sie deren Funktionen und aus welchen Teilen sie üblicherweise besteht. (5.Punkte) (5)

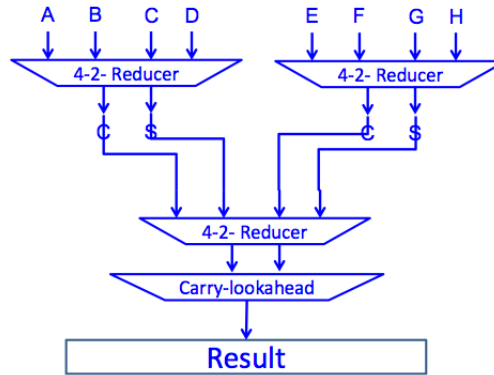
Sagt dafür, dass Interrupt aufgerufen und ausgeführt werden → unterbricht das Hauptprogramm

- PC abspeichern
- Interrupt aufrufen
- Interrupt ausführen (dieser könnte wieder einen Interrupt aufrufen usw.)
- Interrupt enable auf 1 damit bereit für nächsten Interrupt
- PC laden
- Programm läuft weiter

⊗

15. Skizzieren Sie die Struktur eines 4-2 Reducer basierenden Multiplizierers. Was ist sein Hauptvorteil gegenüber einem Array Multiplizierer. (5.Punkte)
(3)

A: Vorteile: Ergebnis noch weniger Schritten?
es schneller



16. Nennen Sie vier Speicher in einer möglichen Memoryhierarchie und beschreiben Sie jeweils deren Eigenschaften. Quantifizieren Sie deren Zugriffszeiten im Verhältnis zum Prozessortaktzyklus.(5.Punkte) (4)

A:

Register 1ns, schnell, flüchtig, kleine Größe, wird für jeden Prozess benötigt

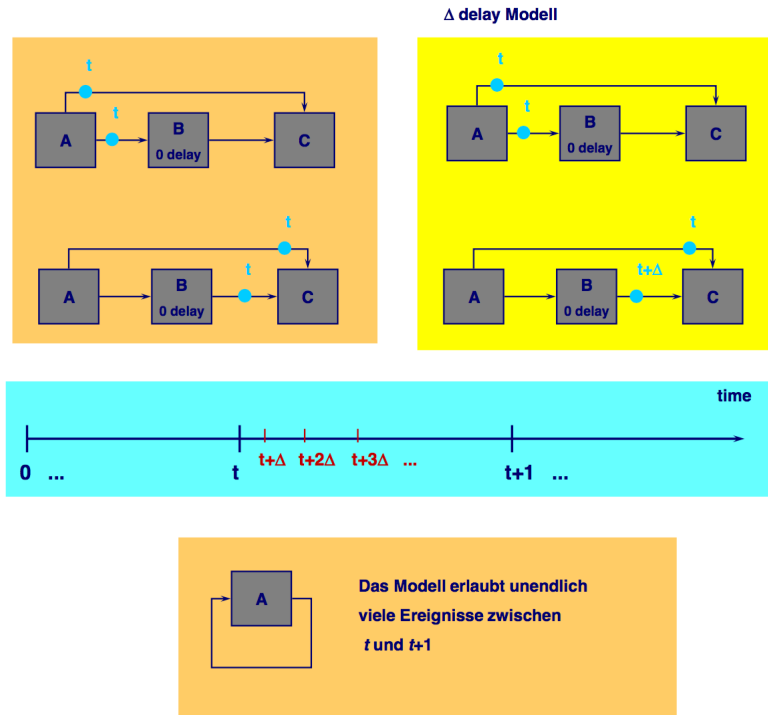
Cache 2ns

Flash nicht flüchtig, sehr Groß, für viele Daten, sollte nicht für jeden Prozess benötigt werden \Rightarrow Copy Zugriffszeiten

HDD ms

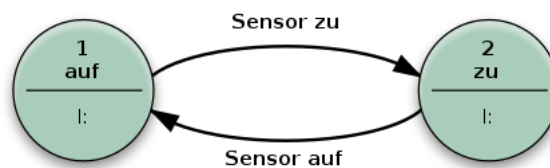
17. Motivieren und beschreiben Sie das δ -Modell von VHDL. (5.Punkte) (4)

A: Zwischen den Zeitpunkten t und $t+1$ für unendlich viele Zeiten liegen.



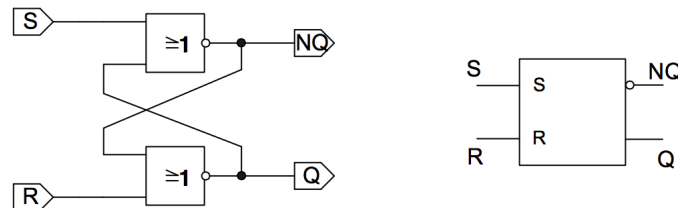
18. Skizzieren und beschreiben Sie Funktion und Aufbau eines Mealy Zustandsautomaten (5.Punkte) (0)

Ein Mealy-Automat ist ein endlicher Automat, dessen Ausgabe von seinem Zustand und seiner Eingabe abhängt. Die Ausgabe erfolgt asynchron.



19. Skizzieren und beschreiben Sie Funktion und Aufbau eines RS-Latches. (5. Punkte) (0)

Ein Latches und Flip-flops besitzen die Eigenschaft genau 1bit zu speichern. Wobei Latches Pegel-gesteuert sind und Flip-Flops Flanken-gesteuert sind. Benutzt man mehrere solcher Bausteine ist es möglich ein Register zuzubauen. Das RS-Latch dient als Ausgangspunkt für alle weiteren Speicherschaltungen (Basis-RS-Latch)



20. Skizzieren und beschreiben Sie Funktionsweise und Aufbau einer Single Instruction Multiple Data (SIMD) Rechnerarchitektur. (5.Punkte) (5)

Eine Instruktion wird vom Speicher an den Controller übergeben. Dieser übergibt die Aufgaben den einzelnen Cores weiter und diese speichern diese an eine vordefinierte Stelle im Speicher.

