

Betriebssysteme Blatt 4

Baran Güner, bg160

Tobias Hangel, th151

18. November 2022

Aufgabe 1

3.5/7

a)

Taktrate: 800 MHz = 800.000.000 Hz

Datenübertragung: 8 MB/s = 8.000.000 Byte/s = 64.000.000 Bit/s

8 * 32-Bit-Worte = 256 Bit

Overhead: 1000 Taktzyklen

Umrechnung mit 2^{20} , das sind Mibinicht Mega
nicht 10^6

1 Zyklus = 7 Takte, 1000 Zyklen = 7.000 Takte

1 Takt = 0,0000000125s, 7.000 Takte = 0,00000875s

$$\frac{256}{64.000.000} = 0,000004s$$

Folgefehler

⇒ Alle 0,0000004s gibt es 0,00000875s lang einen Overhead.

$$\frac{0,00000875}{0,00000875 + 0,000004s} = 0,686$$

diesen Wert
habt ihr nicht

⇒ Bei konstanter Nutzung der Festplatte entstünde 68,6% der Zeit Overhead.

$$0,05 * 0,686 = 0,034$$

⇒ Wenn die Festplatte 5% der Zeit aktiv ist, entsteht 3,4% der Zeit Overhead.

Ah, aber diese 5% sind nicht auf Takte bezogen und nur die RETI hat 7 Takte für Fetch und Execute. Die 5% sind der Anteil der Zeit, der für die korrekte Datenübertragung verantwortlich ist von der gesamten CPU Laufzeit für die 8MB pro Sekunde der Festplatte

1.5/3 für teilweise korrekte Ansätze und die interessante Idee ^_^

b)

Übertragung: 16kb = 128.000 Bits. Benötigt 0,02s

Aktivierung: 1500 Taktzyklen = 10500 Takte = 0,000013125s

Bearbeitung des Interrupts: 500 Taktzyklen = 3500 Takte = 0,000004375s

Aktivierung + Bearbeitung = 0,0000175s

+ Übertragung = 0,0020175s

Während des Datentransfers kann die CPU arbeiten.

$$\frac{0,0000175}{0,0020175} = 0,0086$$

Folgefehler

2/4 für teilweise korrekte Ansätze. Die Aufgabenstellung ist aber auch sehr verwirrend gestellt, da muss es auch Punkte für Lösungen die einfach den falschen Pfad gefolgt sind geben.

⇒ Bei konstanter Nutzung der Festplatte entstünde 0,86% der Zeit Overhead.
 $0,86 * 0,05 = 0,00043$

Wenn die Festplatte 5% der Zeit aktiv ist, entsteht 0,043% der Zeit Overhead.

Aufgabe 2

1/3 für die Mühe

a)

Interruptvektortabelle? Ihr meint wohl Interrupt-Controller IC?

daneben gibt es noch das IVN-Register, also Interruptvektornummer-Register

Der Interrupt Controller teilt durch die IPT der CPU mit, welche ISR ausgeführt werden muss. Diese Service Routine wird in einer bestimmten Anzahl von Taktzyklen abgearbeitet. Ein Taktzyklus besteht aus 7 Takten.

Sobald der Interrupt Controller der CPU die ISR mitgeteilt hat, wird der Eingang /reset aktiviert und der 8-Bit-Zähler wird auf 0 initialisiert. Wenn der Zähler bei 6 angekommen ist, sind 7 Takte, also ein Taktzyklus vergangen. Sobald der Zähler auf 7 steht, überprüft der Interrupt Controller, ob das /INTA Signal durch die CPU aktiviert wurde. Ist dies der Fall, so wurde die Routine beendet. Ist das Signal nicht aktiv, setzt der Controller das Signal von up = 1, down = 0 auf up = 0, down = 1. Sobald der Zähler die 0 erreicht hat, wird erneut geprüft, ob das /INTA Signal aktiv ist. Ist dies nicht der Fall, setzt der Controller das Signal wieder auf up = 1, down = 0 und der Zähler beginnt erneut hoch zu zählen...

Dies wird wiederholt, bis das /INTA Signal aktiviert wird.

dafür gibt es doch schon E, s0 und s1. Der Zähler ist fürs Inkrement. und dekr. da

Es ging in der Aufgabenstellung darum festzustellen, ob eine ISR gerade aktiv ist.

b)

Eine solche Situation könnte vorliegen, wenn ein Softwareinterrupt oder eine Trap im Prozessor passiert. Dies soll aber laut Aufgabenstellung vernachlässigt werden.

Problematisch könnte auch sein, dass der Interrupt Controller durch diese Vorgehensweise nicht weiß, ob die CPU gerade eine ISR durchführt, die eine niedrigere Priorität hat, als die ISR, die der Controller als nächstes schicken möchte.

Der Interrupt Controller muss sich merken, welche Priorität der letzte Interrupt, den er an die CPU gesendet hat, hatte. Nach jedem Zyklus des 8-Bit Zählers subtrahiert der Interrupt Controller den Inhalt vom Register PR, der zur aktuell laufenden ISR gehört vom Inhalt des Registers PR, der zum IVN Register des anliegenden Interrupts mit der höchsten Priorität gehört. Wenn das Ergebnis negativ ist, der anliegende Interrupt also eine höhere Priorität hat, wird ein neues INT Signal an die CPU geschickt. Ist dies nicht der Fall, wird der Vorgang nach dem nächsten Zyklus wiederholt.

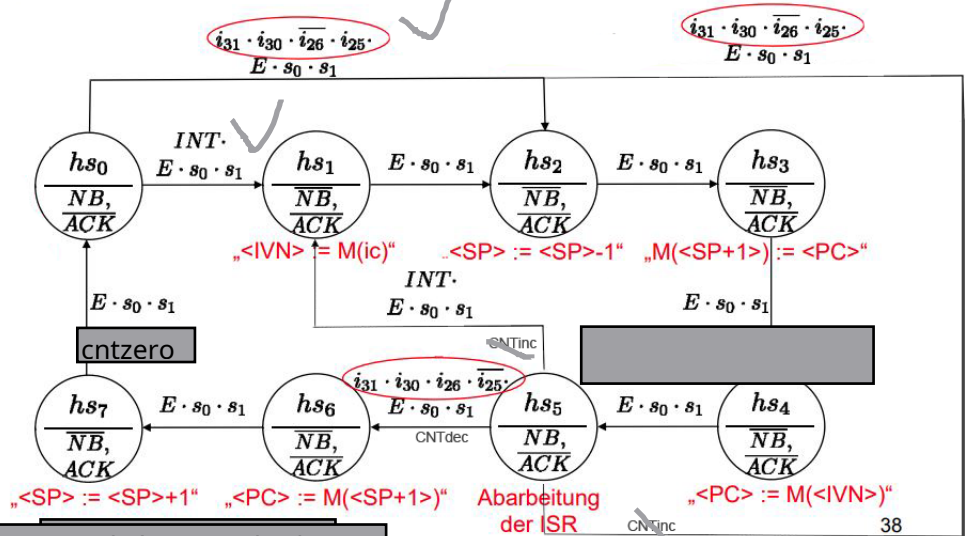
geht in die richtige Richtung

die Kontrollogik braucht 8 Taktzyklen, aber eine ISR selbst braucht so viele Taktzyklen, wie sie Befehle hat

1.5/4 für die Mühe

Aufgabe 3

4.5/6



was wenn ein Interrupt durch einen Interrupt unterbrochen wurde und der letztere gerade abgearbeitet wurde und der erstere noch abgearbeitet werden muss?

Das ist ein Moore Automat, also gehört CNTINC in die Knoten, weil es Output ist

38