

Aufgabe 1

1. Beim Polling werden die Interrupt-Quellen regelmäßig zyklisch abgefragt, ob eine Anfrage anliegt. Bei reinen Interrupts kommen von den Komponenten aktiv die Interrupts, die den Prozessor unterbrechen.
2. Es gibt beim Polling zwei Varianten: Entweder wird nach der Bearbeitung eines Interrupts in der ursprünglichen Reihenfolge die nächste Komponente abgefragt, oder es wird in der Liste der Komponenten wieder von vorn begonnen. In Variante 1 sind die Komponenten gleichberechtigt, in Variante 2 haben die Komponenten Prioritäten gemäß ihrer Stelle in der Abfrageliste.
3. Der Prozessor bekommt bei einem Interrupt die jeweilige Nummer des Interrupts mitgeteilt. In der IVT (Interrupt Vector Table) ist jeder Nummer eine Adresse zugeordnet, wo die auszuführende ISR (Interrupt Service Routine) zu finden ist. Die ISR sichert benutzte Register und setzt den Interrupt Enable Bit. Dann wird das Anliegen des Interrupts behandelt, bzw. die benötigten Daten und die Erlaubnis zur Bearbeitung an die zuständige Stelle weitergeleitet. IVT und ISR werden vom Betriebssystem festgelegt.
4. Ein Interrupt kann von einem Interrupt mit höherer Priorität unterbrochen werden.
5. siehe Skizze
6. Die Daisy-Chain ist eine hardwareseitige Lösung, die den Prozessor nicht unnötig belastet und gleichzeitig den Interruptquellen Prioritäten zuweist. Der Prozessor hat somit nur minimalen Aufwand mit Interrupts.
7. Löst eine oder mehrere Quellen einen Interrupt aus, so erhält die CPU ein Signal am IRQ-Pin. Wenn das Interrupt Enable Bit gesetzt ist, wird der Interrupt gestattet und entsprechend das IACK Signal gesetzt. Dieses Signal geht der Reihe nach durch die Daisy-Chain, wobei die erste Quelle, die einen Interrupt ausgelöst hat, das Signal für die nachfolgenden Quellen sperrt. So wird der Interrupt mit der höchsten Priorität ausgeführt.
8. Daisy-Chains sind nicht fair, die Reihenfolge der Quellen entspricht ihrer Priorität.

Zeitmessung

1. Der Aufruf der Webseite im Browser verursacht Interrupts und Prozessorlast. Angefangen bei der Mausbewegung, Grafikkarte, dem Browser selbst und der Netzwerkkarte unterbrechen diverse Prozesse die CPU beim Berechnen unseres Programmes. Dadurch verlängert sich die Laufzeit.
2. Der Interrupt wird ausgelöst. Momentane Arbeit wird unterbrochen. Art des Interrupts wird festgestellt (Software-/Hardware-Interrupt). Interrupt Enable-Bit wird geprüft. Interrupt wird zugelassen. PSW und PC werden auf dem Stack gesichert. Interrupt Enable Bit wird zurückgesetzt. ISR-Adresse wird ermittelt. Benutzte Register werden auf Stack gesichert. Interrupt Enable Bit wird wieder gesetzt. ISR wird ausgeführt. ISR wird beendet. PSW und PC werden wieder hergestellt und zum ursprünglichen Programm zurückgekehrt.

Aufgabe 2

;Bearbeiter: Bjoern Oke Maas, Mihai Renea
;Tutorium: Jonas Cleve, Do. 14 Uhr

global collatz

section .text

```
collatz:
mov rax, -1          ; do-while-Schleife, also
                    ; Zaehler ab -1

_loop:
inc rax              ; Zaehler +1
lea rsi, [rdi*2+rdi+1] ; rsi = rdi*3+1
shr rdi, 1           ; rdi = rdi/2

cmovc rdi, rsi       ; Ergebnis aus rsi wird genommen
jnz _loop            ; falls CF=1, weil dann rdi ungerade war
ret                  ; _loop falls ZF=0, weil dann rdi!=1 war
```

1.5)

