

ALBERT-LUDWIGS-UNIVERSITÄT FREIBURG

BETRIEBSSYSTEME

Blatt 02



Students:

Julian Polzer jp390, David Janzen dj57

Tutor:

Sophronius Eusebius
Hieronymus und bitte mit
vollem Namen ansprechen

1 Aufgabe

Befehl | Treiber:

1. STOREIN ACC SP i | ACCLd, IRd, SPDd

Der Befehl speichert im SRAM an Stelle $ACC + i$ den aktuellen Wert von SP ab. i steht im Instruktionsregister. Dann legt man den ACC mit ACCLd auf dem linken und i mit IRd auf dem rechten Bus. Die ALU rechnet dann $ACC + i$ und SP wird über SPDd zum SRAM gebracht.

-0.25 ALUAd
vergessen

2. MOVE IN2 ACC | IN2Ld, 0Rd, ALLUId

Speichert den Wert des Registers IN2 im ACC ab. IN2Ld legt die Informationen auf den linken Datenbus. Mit 0Rd verodern in der ALU und dann zum ACC bringen über ALLUId.

-0.25 es war nach dem Weg mit den wenigsten Treibern

3. ADD ACC IN1 | Nicht Möglich

Weil es keine Möglichkeit gibt den ACC auf den Rechten Datenbus zu legen und zeitgleich den IN1 auf dem linken

-0.5 doch z.B. über In1Dd und DRd

-1 keine d)

2 Aufgabe

1. Für Push muss der Inhalt des ACC auf dem Speicher geschrieben werden. Dafür muss SP als Adresse ebenfalls an den Speicher angelegt sein. Im Anschluss darauf muss SP um eines herabgesetzt werden damit es wieder auf das Feld $n-1$ zeigt. Dafür werden die Befehle 'STOREIN ACC SP i' (mit $i = 0$) und im Anschluss darauf den Befehl 'SUBI SP i' (mit $i = 1$).

Für Pop muss der Inhalt des SP an Stelle n in den ACC geschrieben werden und im Anschluss der StackPointer um eins angehoben werden. Dafür werden die Befehle 'LOADIN SP ACC i' (mit $i = 1$) und 'ADDI SP i' (mit $i = 1$) verwendet.

-0.25 der SP sollte immer auf eine freie Speicherzelle zeigen

2. Die Befehle INT i und RTI existieren, um die Verwendung von Systemcalls und Interrupts von dem normalen usermodus der RETI zu trennen. Der Grund dafür ist, dass Systemcalls höhere Berechtigungen haben, als sie der Nutzer sonst hätte.

beste Antwort bisher.

3 Aufgabe

Da die Aufgabe seltsam gestellt ist, gibt es hierfür so oder so volle Punkte.

aber die RETI macht das nicht, aber andere Architekturen machen sowas