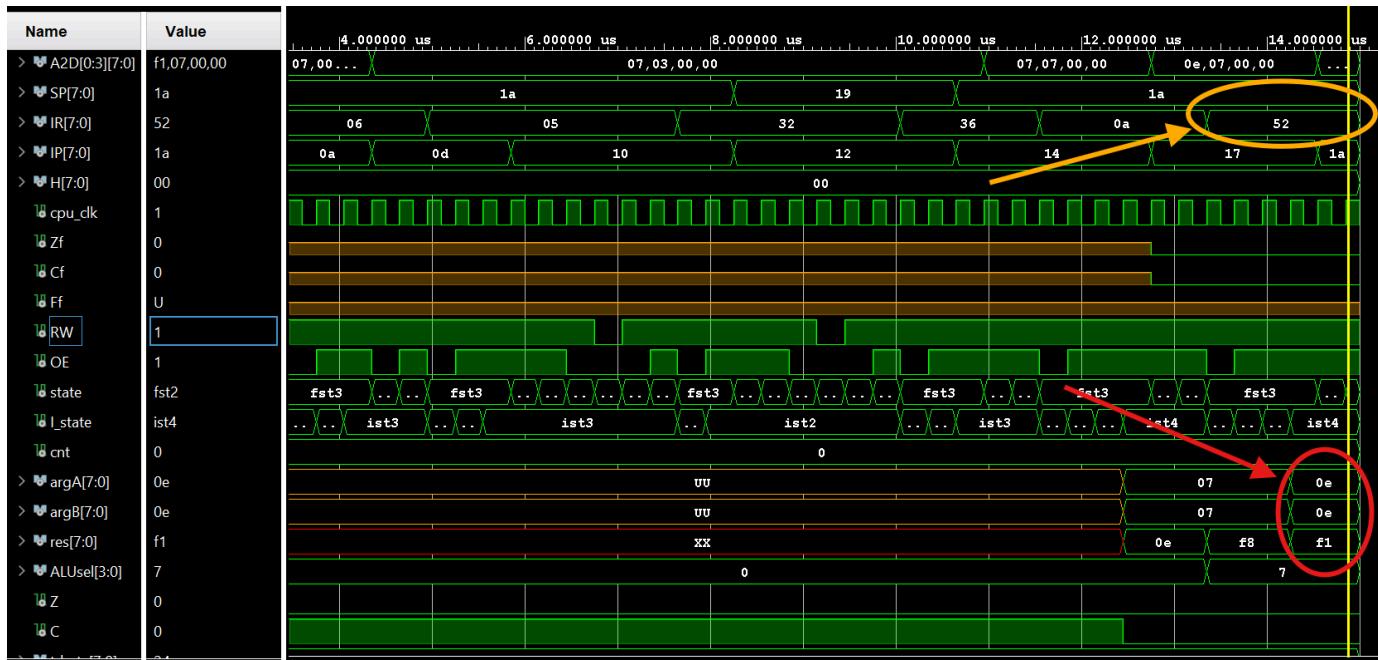


1)

Det ligger i koden, men instruktionen blev tilført i ALU, og derefter i top\_w\_proc2.

Instruktionen er egentlig bare at sige NOT argA. Opcode for denne instruktion er 52, eller "0x52"

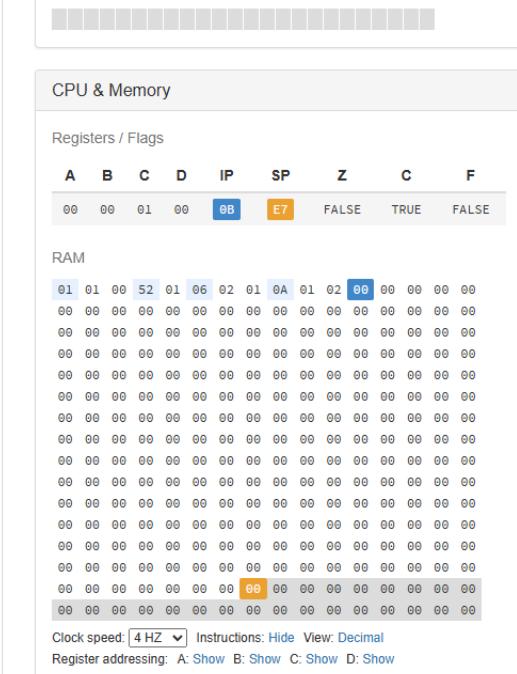
2)



Vi kan se i den orange cirkel at vi når til 0x"52" i intern ram, og i den røde cirkel ser vi argA (0e) og resultat (f1). f1 er not af 0e. Vi kan bemærke at både RW og OE er 1, samt også at instruktionen først skrives når ist4 er igang.

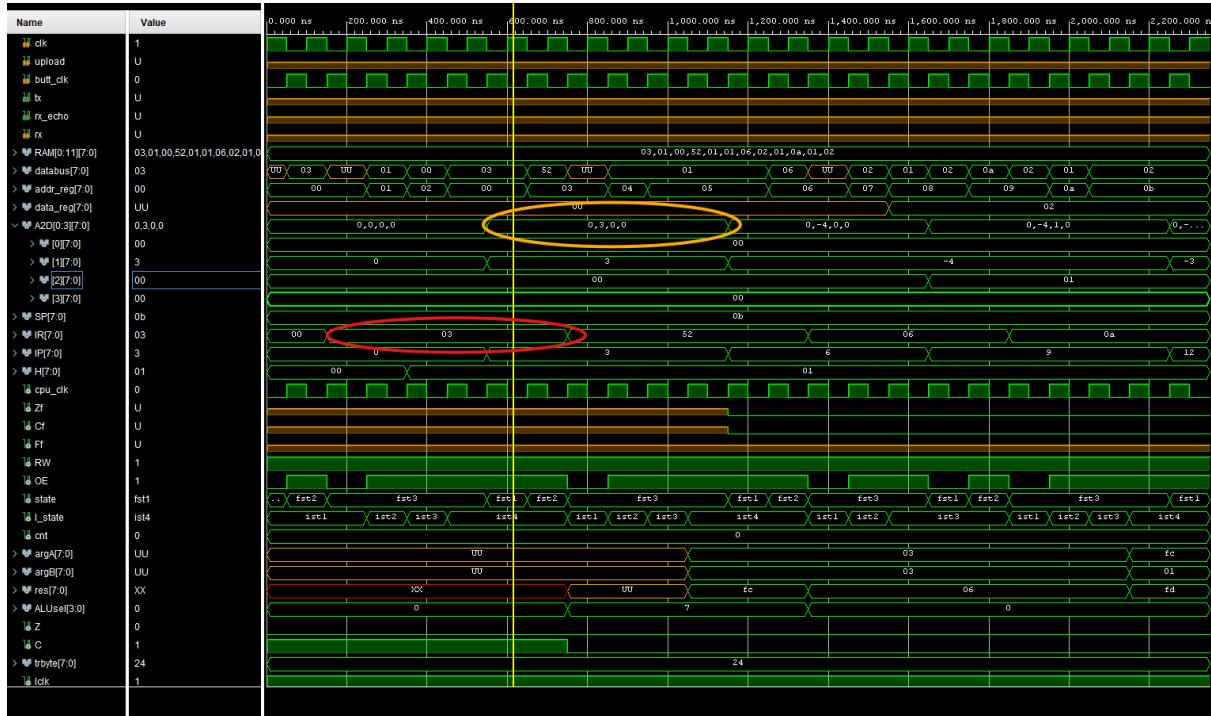
3)

```
MOV B, A      ; Write fra register  
NOT B        ; Negation  
MOV C, 1      ; Flyt konstant til register  
ADD B, C      ; Adder register
```

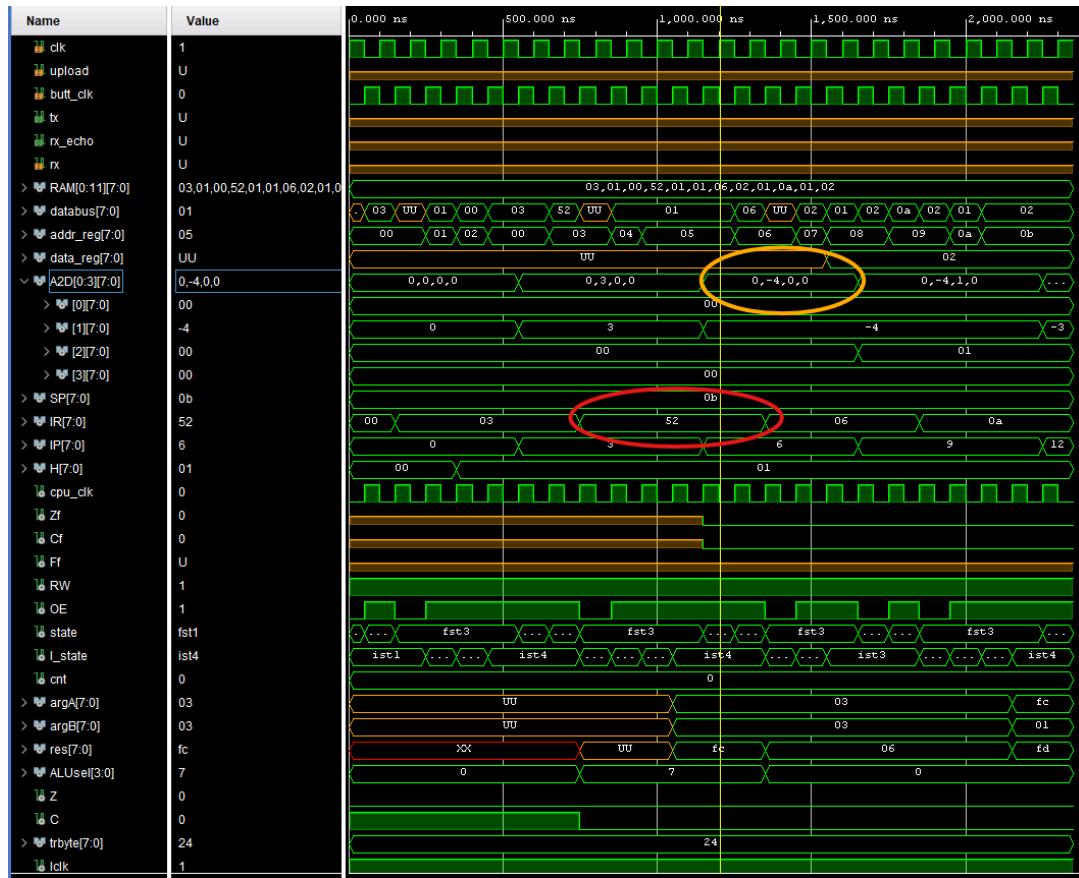


4)

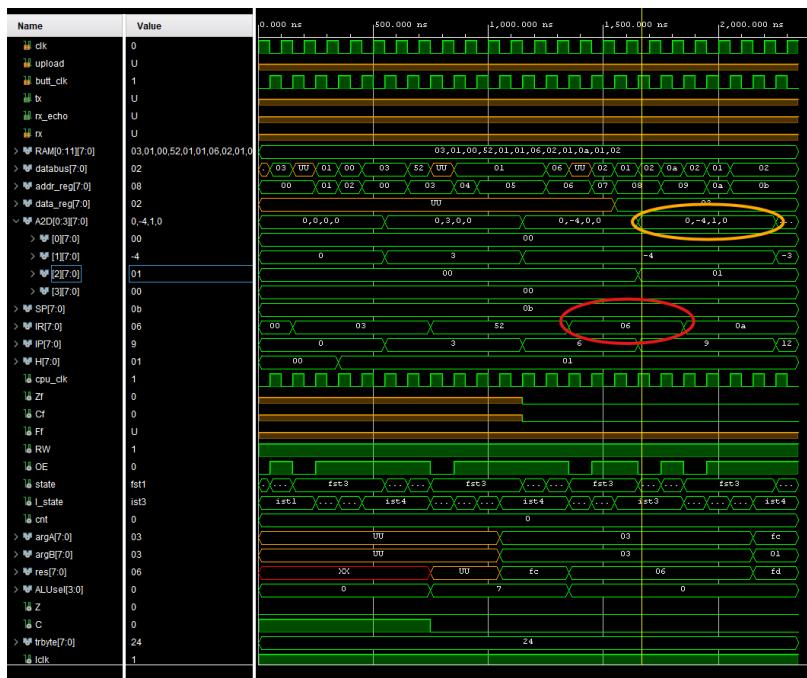
linje 1:



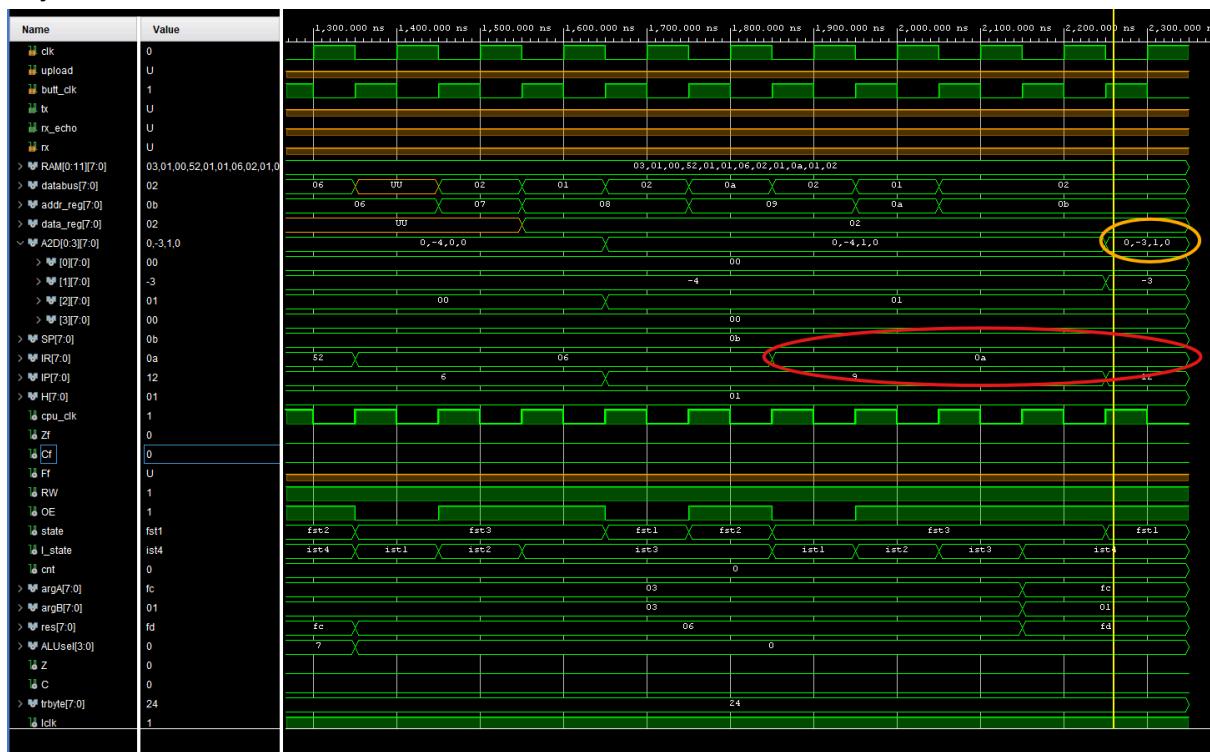
linje 2:



Linje 3:



Linje 4:



5)

impl interrupt flags

PC skal peje på hvor interrupt routine ligger og gemme (context save) hvad den pejede på før, samt register indhold

tjek for høj signal på interrupt port [i opcode\_fetch();] -> sæt flag -> næste cycle gem PC på stack og sæt PC til ISR adresse