

Bezettingsadministratie der trommelpagina's.

Aan elke trommelpagina kennen we twee constanten toe, nl. a en s : dit zijn een-eenduidige functies. De constante a "nummert" de trommelpagina's in volgorde van opklimmend beginadres, de constante s "nummert" de trommelpagina's in de sequens, waarin de ~~beginwoorden~~ beginwoorden van de pagina's de koppen passeren.

Voorlopig stellen we ons op het standpunt, dat de hele trommel voor pagina's ter beschikking staat. (Dit zal niet het geval zijn, maar zulks kunnen we opvangen door een aantal trommelpagina's permanent bezet te verklaren, waarna we ze rustig voor andere doeleinden kunnen gebruiken.)

Als functie van a is het beginadres gegeven door

$$\text{beginadres}(a) = 513 * a \quad ;$$

tussen opeenvolgende trommelpagina's zit dus een ongebruikt woord. Verder geldt

$$0 \leq a \leq 1021 \quad ,$$

voor $a = 1022$ en $a = 1023$ is geen ruimte meer op de trommel. De laatste drie woorden op de trommel worden eveneens niet gebruikt.

Nu merken we op, dat $513 * 513 \equiv 1 \pmod{1024}$; alle adressen $\equiv \pmod{1024}$ zijn tegelijkertijd accessibel. Hieruit volgt, dat verhoging van s met 1 naar komt op verhoging van a met 315, gereduceerd $\pmod{1024}$.

Voegen we aan de pagina met $a = 0$ ook $s = 0$ toe, dan vinden

$$a(s) = \text{rem}(513 * s, 1024).$$

Omgekeerd vinden, dat verhoging van a met 1 de s met 513 verhoogt (modulo 1024) dus geldt

$$s(a) = \text{rem}(513 * a, 1024).$$

Ok s voldoet primair aan de ongelijkheid $0 \leq s \leq 1023$; we moeten ons even afvragen welke s -waarden niet voorkomen; dit zijn

$$\begin{aligned} s(1022) &= 1022 & \text{en} \\ s(1023) &= 511 . \end{aligned}$$

Deze nonexistenten bladzijden laten we vrolijk meehobbelen, we zullen ze (zijnde non-disponibel) als permanent bezet noteren.

Voorts is s als functie van het beginadres ten duidelijkste gegeven door

$$s(\text{beginadres}) = \text{rem}(\text{beginadres}, 1024);$$

alle tot dusverre bekeken functie's zijn dus door schoon schuiven over 9 plaatsen, een enkele additie en een collatie met 1023 onmiddellijk uit te rekenen.

Aan deze pagina's gaan we bits toekennen uit een boolean array $\text{TBB}[0:39, 0:25]$; (de afkorting TBB komt van Trommel Bezettings Bits). De bits van dit array gaan we afbeelden op de bits $d[25]$ t/m $d[0]$ van de woorden van het woordarray $\text{TBW}[0:39]$, 40 woorden, in het geheugen te bergen op 40 consecutieve geheugenplaatsen in volgorde van oplopend subscript. Van deze 40 woorden is het tekenbit $d[26]$ permanent gelijk aan 1.

Voor de bits van TBB geldt, dat

- a) $\text{TBB}[i, j]$ afgebeeld wordt op bit $\equiv d[25 - j]$ uit woord $\text{TBW}[i]$
- b) $\text{TBB}[i, j]$ toegevoegd is aan de trommelpagina met $s = 26 * i + j$
- c) $\text{TBB}[i, j] = 1$ betekent, dat de pagina bezet is
 $= 0$ betekent, dat de pagina vrij is.

De bits TBB[19,17] (dwz. $s = 511$), TBB[39,8] (dwz. $s = 1022$) en TBB[39, 10], TBB[39, 11]...TBB[39, 25] (dwz. $s = 1024, 1025, \dots, 1039$) zijn wegens non-existent permanent = 1, evenals de TBB's, die beantwoorden aan stukken van de trommel, die voor andere doeleinden bestemd zijn.

(In het volgende is de constante $B25 = 2^{125} = 33554432$.)

We zullen de TBB's op drie manieren gebruiken.

a) Om een bezette trommelpagina met $s = 26 * i + j$ vrij te geven.

In een register plaatsen we $B25 * 2^{(-j)}$ en met een subtractieve uitopdracht voeren we uit

$$TBW[i] := TBW[i] - B25 * 2^{(-j)}.$$

b) Om een vrije trommelpagina met $s = 26 * i + j$ te bezetten; dit gaat analoog met een additieve uitopdracht:

$$TBW[i] := TBW[i] + B25 * 2^{(-j)}.$$

De weg van beginadres naar s gaat met een simpele collatie, om uit s de i en de j te isoleren, kost ons wel een deling. De tekencijfers van de woorden TBW zijn met zorg =1 gekozen, opdat de preferentie voor de min nul ons geen parten speelt. Als wij deze vrijgave en bezetting additief spelen, dan moeten we er wel zeker van zijn, dat de pagina van te voren als bezet, resp. vrij genoteerd stond; anders zou overloop de boel immers hopeloos bederven!

c) Om een geschikte vrije trommelpagina te zoeken. Als we nl. dumpen, dan zijn we vrij in de keuze van de trommelpagina. Als zich in het startmagazijn van de trommel-transporten nog een startopdracht bevindt, dan kunnen we onderzoeken, bij welke waarde van s deze is afgelopen. Verderderen we (gereduceerd modulo 1024) deze s met de "safety marge", die Electrologica ons op moet geven, opdat de volgende startopdracht nog zonder verlies van een omwenteling gehaald wordt, dan hebben we de ideale s , ware het niet, dat we nog moeten onderzoeken, of de bijbehorende trommelpagina wel vrij is. (Als er in het startmagazijn van de trommel nog maar 1 startopdracht staat, kunnen we in verband met de minimum tijdsduur van het nu volgende onderzoek-programma uit voorzichtigheid s nog wel met een of andere constante verhogen, extra wel te verstaan. Als het startmagazijn laag is, is elke s even goed, want dan hebben we de stand van de trommel volledig uit het oog verloren.) Het array TBB gaan we nu gebruiken om de eerste vrije trommelpagina (met " $s \geq s$ ideaal", alles modulo 1024) te zoeken. Daartoe splitsen we s ideaal in

$$s \text{ ideaal} = 26 * i + j$$

en voeren met deze waarden van i en j als beginwaarden het volgende programma uit.

$A := TBW[i];$

schuif A schoon over j plaatsen naar links;

if $A \neq 0$ then

begin normeer A (het aantal slagen komt in B);

$j := j + B$

end

else

begin $i0 := i;$

 volgend woord: $\text{modhoog}(i, 40);$

$A := TBW[i];$

if $A = 0$ then begin if $i = i0$ then goto trommel vol else goto volgend woord

end;

 normeer A (het aantal slagen komt in B);

$j := B$

end

De gekozen pagina is nu gegeven door de eindwaarden van i en j (door $s = 26 * i + j$), waarmee het gestelde doel bereikt is.