

toets 2000-5

Een microprocessor in een zeer eenvoudige vorm bestaat uit een datapad, een stuur-eenheid en een zogenaamde *cache*. In wezen is zo'n *cache* niets anders als "geheugen op de chip". Hiermee kan voorkomen worden dat men voor elke instructie de code en data uit het werkgeheugen moet halen, dat zich op een andere chip bevindt. De stuur-eenheid is een brok logica die de instructies decodeert en hieruit signalen afleidt die de eenheden (*modules*) in het datapad activeren en het verkeer tussen de cache en het werkgeheugen regelt. Het datapad bevat dus modules die bewerkingen kunnen uitvoeren of woorden kunnen onthouden. Dat onthouden gebeurt in *registers*. Een register is een aantal geheugencellen in een rij geplaatst, en elk voorzien van dezelfde klok- en stuursignalen. Het aantal cellen in zo'n rij is meestal gelijk aan de woordlengte. De bewerkingen die in een datapad uitgevoerd worden, zijn meestal van combinatorische aard: rekenkundige en logische bewerkingen. Ook hier geldt voor dat ze in rijen van (nagenoeg) identieke cellen geplaatst worden en ook hier is het aantal gelijk aan de woordlengte.

De communicatie tussen de eenheden vindt plaats via bussen. Een bus is een groep draden (metaal-interconnecties) die precies dezelfde modules verbinden. Het aantal draden in een bus is weer gelijk aan de woordlengte. Een datapad ziet er op de chip dus uit als een rechthoek met daarin een zich in één richting herhalend patroon. Zo'n patroon heet een *bitslice*. Het aantal bitslices is dus ook gelijk aan de woordlengte. Elke bus is binnen zo'n bitslice vertegenwoordigd met precies één van zijn draden. Zo'n draad moet alle modules die via die draad communiceren passeren. In het uiterste geval is zo'n draad dus net zo lang als het datapad, namelijk als de twee uiterste modules met elkaar moeten communiceren.

Microprocessors zijn meestal synchrone digitale systemen. Dat wil dus zeggen dat alle bewerkingen "gesynchroniseerd" worden door een klok. Bij een zogenaamde *rise*-processor wordt er binnen elke klokperiode precies één instructie (of één *pipe line stage*) afgewerkt. De snelheid van de klok moet zodanig zijn dat alle operaties netjes afgehandeld kunnen worden: het vrijgeven van de bus, de propagatie van de signalen van bronregister naar het bewerkingsmoduul, de bewerking in dat moduul, het transporteren van het resultaat naar het ontvangende register, en het inlezen in dat register. Natuurlijk wil men de klok een zo hoog mogelijke frequentie geven, maar de hierboven genoemde reeks legt hier een bovengrens aan. Hierbij moet men met de slechtst mogelijke situatie rekening houden: de langzaamste bewerkingen en de langste propagatietijden. We willen de bijdrage van het data-transport in het slechtste geval bepalen.

Het datapad is zo'n 4 cm lang. De langste busdraad stellen we daarom op die lengte. De totale weerstand van de draad is $4 k\Omega$ en de totale draadcapaciteit is $1 pF$. Verder veronderstellen we dat deze draad met 9 modules verbonden is en dat deze connecties equidistant zijn. Elke connectie belast de draad met $100 fF$, ongeacht of de module een ontvanger is of niet. De equivalente weerstand van een minimale transistor is $6 k\Omega$.

Vraag 1: Met hoeveel tijd moet bij het bepalen van de hoogste klokfrequentie rekening gehouden worden voor het transporteren van data tussen datapadmodules?

Vraag 2: Wat kan men doen om dit aandeel zo klein mogelijk te maken?

Vraag 3: Wat is het minimum voor dit aandeel?
Geef een realistische aanwijzing om dit minimum (vrijwel) te halen!