

Opdracht week 6 – 4-bit 2's complement comparator

Inleiding

In de digitale techniek worden getallen opgeslagen in het binaire talstelsel. Reken-schakelingen zijn eenvoudig te ontwerpen.

Een *comparator* is een schakeling die twee binaire getallen vergelijkt en aangeeft hoe de verhoudingen liggen. Zo geeft de schakeling aan of de twee getallen gelijk zijn, of dat het ene getal groter is dan het andere getal. Natuurlijk zijn meer vergelijkingen mogelijk zoals ongelijk, groter of gelijk en kleiner of gelijk.

Het hart van de comparator is een aftrekschakeling. Met deze schakeling is mogelijk om uitspraken te doen over twee getallen:

$A - B = 0$ getallen zijn gelijk

$A - B > 0$ A is groter dan B

$A - B < 0$ A is kleiner dan B

Een aftrekschakeling is te bouwen uit een optelschakeling, immers $A - B = A + (-B)$.

De opdracht is om een 4-bit 2's complement comparator te ontwerpen op basis van de eerder ontworpen 4-full adder.

Leerdoelen

De leerdoelen van deze opdracht zijn:

- Ontwerpen van een rekenschakeling voor 2's complement getallen.
- Ontwerpen van een schakeling voor testen of een getal groter is dan een ander getal.
- Ontwerpen van een schakeling voor testen of een getal kleiner is dan een ander getal.
- Ontwerpen van een schakeling voor testen of een getal gelijk is aan een ander getal.
- Invoeren, simuleren en testen van de ontworpen schakeling.

Opdrachten

De volgende opdrachten moeten gedaan worden. Eerst moet het project ingericht worden.

- a) Haal van BlackBoard het zip-bestand `comparator.zip` binnen en pak het uit in `H:\QUARTUS\INLDIG`. Het zip-bestand bevat het Quartus-project `comparator`.

- b) Ontwerp de comparator-schakeling. Het ontwerpen van een is-gelijk-schakeling is niet zo lastig, het ontwerpen van groter-dan en kleiner-dan wel. Ga met meerdere mensen aan de slag, overleg over hoe de schakeling moet worden opgebouwd. De *mappings* van de ingangen en uitgangen staan in onderstaande tabel:

A3	->	SW7	B3	->	SW3
A2	->	SW6	B2	->	SW2
A1	->	SW5	B1	->	SW1
A0	->	SW4	B0	->	SW0

AltB	->	LEDG0	(A less than B)
AeqB	->	LEDG1	(A equals B)
AgtB	->	LEDG2	(A greater than B)

In het schema zijn de signalen S3, S2, S1, S0 en C geplaatst die resp. de vier sombits en de carry-out representeren. Plaats in het schema zelf de signalen N en V. De simulator toont deze in de waarheidstabel.

- c) Simuleer de 4-bit 2's complement comparator met behulp van ModelSim. De bijbehorende simulatiescript heet `tb_comparator.do`.
- d) Test de schakeling met behulp van het DE0-ontwikkelbord.

Opmerkingen

Merk op dat er zes *relationele operatoren* zijn: $=$, \neq , $<$, $>$, \leq en \geq . Je hoeft er maar twee te ontwerpen, de overige zijn af te leiden uit de eerste twee.

Bij het vergelijken van 2's complement getallen komen de overflow (V) en de negative (N) flag in beeld. Stel een tabel op met alle combinaties van N en V en doe een uitspraak over wat de combinatie voorstelt. Bijvoorbeeld het feit dat V logisch 1 is wil zeggen dat het *resultaat* niet geldig is maar het zegt *wel* wat over de verhoudingen van de twee getallen.

Reken als voorbeeld maar $+7 - (-7)$ en $-7 - (+7)$ en $+3 - (-2)$ uit.

N	V	relationele operatie