

***Professionele Bachelor in de Elektronica-ICT***

1 EO

**OPO DIGITALE TECHNIEKEN (LAB)**

## INHOUDSOPGAVE

<b>1 SIMULATIE MET MICRO-CAP .....</b>	<b>1.1</b>
1.1. INSTELLEN EURO SYMBOLEN .....	1.1
1.2. SCHEMA TEKENEN .....	1.5
1.3. SIMULATIE .....	1.15
1.4. AFDRUKKEN .....	1.17
1.5. OPSLAAN .....	1.17
1.6. SCHAKELING AFSLUITEN .....	1.17
1.7. PULSTELLER .....	1.18
1.8. MEER LOGICA .....	1.22
1.9. VERTRAGING .....	1.24
1.10. INITIALISATIE VAN DE FLIPFLOPS BIJ OPSTARTEN .....	1.26
1.11. MIXED MODE .....	1.29
1.12. HET GEBRUIK VAN ANIMATIE-OBJECTEN .....	1.34
1.13. STAP PER STAP SIMULEREN .....	1.36
1.14. INTERESSANTE WEETJES .....	1.41

**Druk alle figuren af en zorg ervoor dat op elke afdruk van een schema of een simulatie je naam, klas, groepsnummer en datum voorkomt.**

Dit voeg je toe d.m.v. de knop tekst



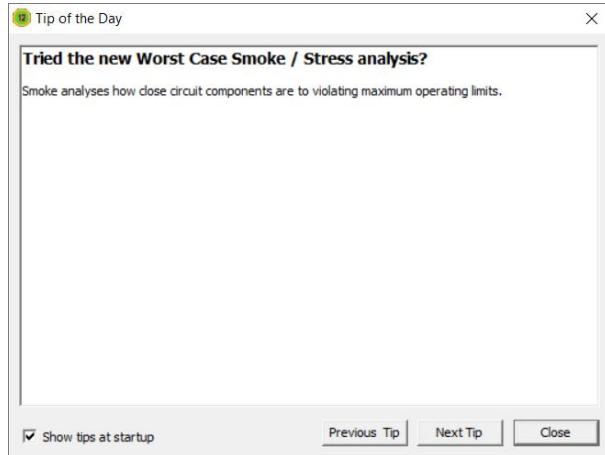
© Claus Dirk

De screenshots zijn gebaseerd op Micro-Cap 12.2.0.3 (64 bit)

**Installeer de volledige versie van Micro-Cap 12 (= MC12) NIET in een schrijfbeveiligde map zoals Program Files. MC12 schrijft immers naar zijn installatiefolder; deze folder moet dus schrijfbaar zijn. Een goede locatie is bijvoorbeeld de defaultwaarde C:\MC12.**

## **1 SIMULATIE MET MICRO-CAP**

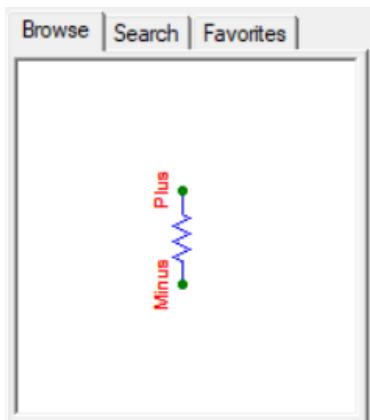
Als je Micro-Cap opstart, word je een ‘Tip of the Day’ aangeboden. Je kan deze telkens lezen om iets extra bij te leren. Een voorbeeld zie je in de figuur hieronder.



Nadien sluit je dit scherm door op ‘Close’ te klikken.

### **1.1. Instellen Euro symbolen**

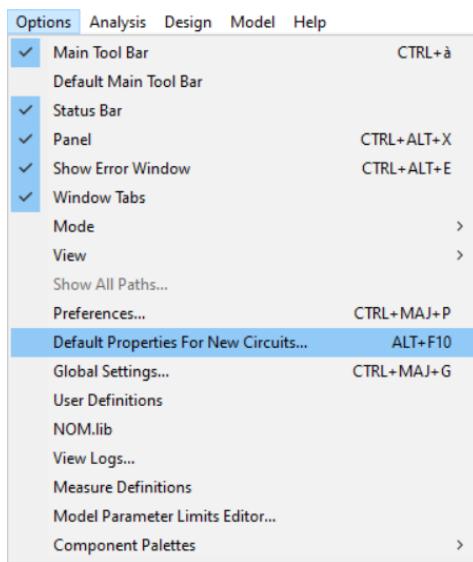
In het scherm van Micro-Cap zie je linksboven het symbool van een weerstand staan. Deze is voorgesteld in de Amerikaanse symboliek (ANSI). Zie figuur hieronder.



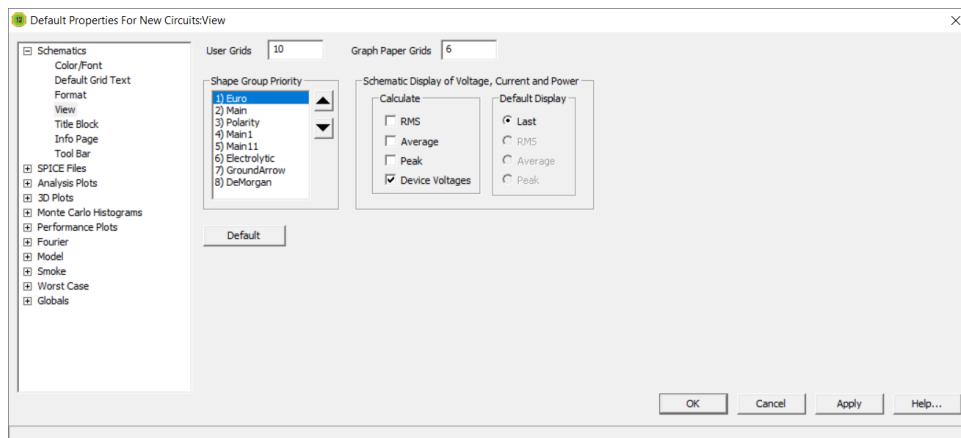
Wij moeten in België volgens de normen de rechthoekige voorstelling voor een weerstand gebruiken; dit zijn de **IEC-symbolen**. Ook in Europa en andere delen van de wereld worden deze symbolen gebruikt.

In Micro-Cap noemt dit de ‘**Euro-symbolen**’. Stel de juiste symbolen eerst in volgens onderstaande werkwijze!

Kies uit het hoofdmenu *Options/Default Properties For New Circuits...* of druk ALT+F10.



Kies bij *Schematics* de optie *View* en zet daar Euro op 1). Je doet dit door eerst te klikken op '2) Euro' en dan op de pijl naar boven . Bevestig dit scherm met OK (of sleep 'Euro' naar de bovenste plaats).



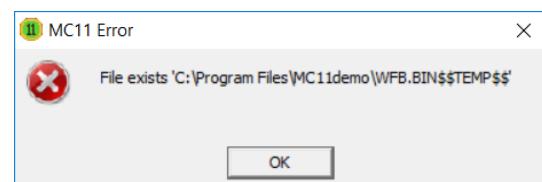
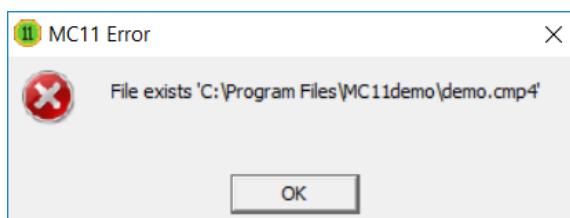
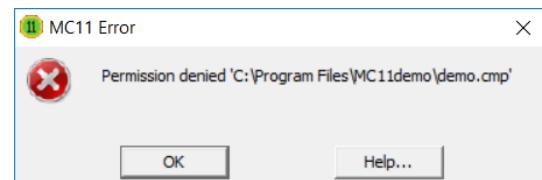
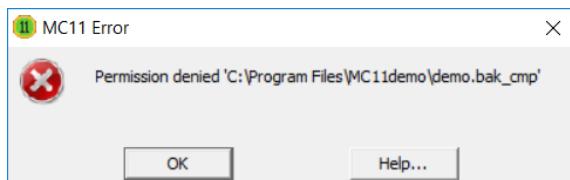
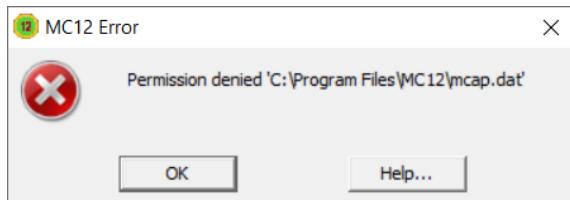
Je merkt dat aan de cursor nog steeds het foutieve symbool hangt met de 'zigzag'-lijn. Om deze aanpassing te laten werken, moet je het programma sluiten en opnieuw opstarten. Je merkt dat na het herstarten het symbool voor de weerstand een rechthoekige vorm heeft. De weerstand ziet er nu zo uit:



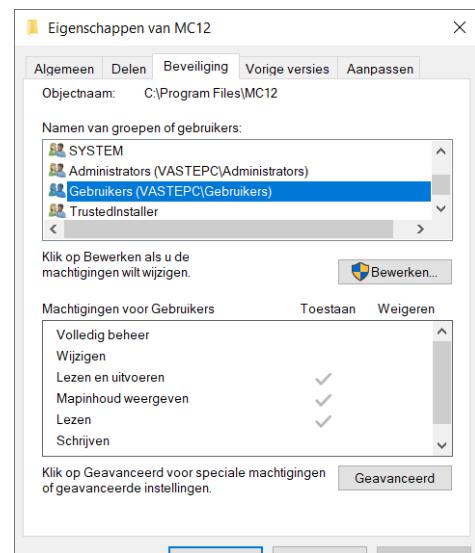
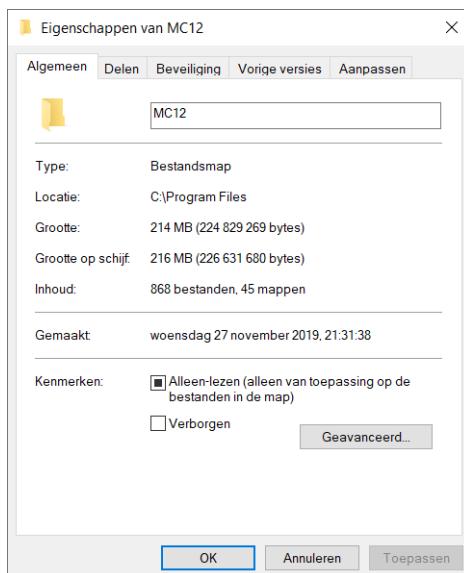
**De symbolen worden enkel aangepast als nog geen componenten geplaatst zijn!**

**Opmerking:** dit beperkt zich tot de eenvoudigste symbolen. Later zal je merken dat bijvoorbeeld digitale poorten met 3 ingangen het nog altijd de Amerikaanse notatie (=ANSI-symboolen) blijft!

**Indien je geen fouten krijgt**, mag je naar ‘1.1.2 Schema tekenen’ gaan! **Indien je op je persoonlijke pc of laptop de volgende foutmelding(en) krijgt** (bevestig telkens met OK), dien je de rechten van de gebruikers van de map waar je Micro-Cap installeerde aan te passen:

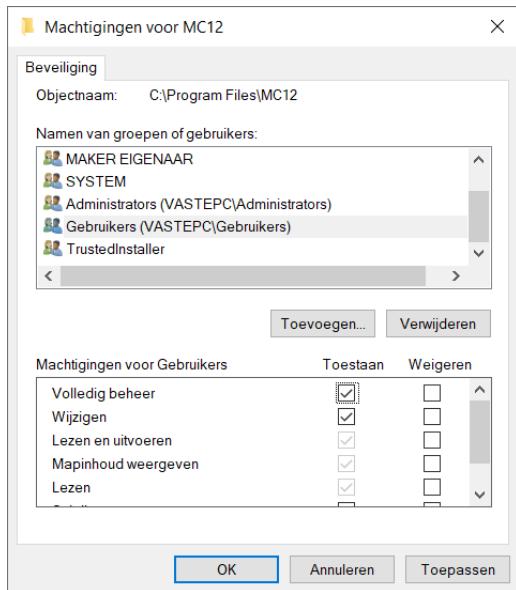


Ga hiervoor met Windows Verkenner naar de Micro-Cap-map. In dit voorbeeld is dit C:\Program Files\MC12. Druk met de rechtermuisknop (RMK) op deze map en selecteer ‘Eigenschappen’. Ga naar het TAB-blad ‘Beveiliging’ en klik op ‘Gebruikers’.



Klik op **Bewerken...**

Selecteer dan ‘Gebruikers’ en vink ‘Toestaan’ bij ‘Volledig beheer’ aan.



Bevestig met OK. (2 keer)

Controleer in Micro-Cap of de fout nu opgelost is.

Indien de weerstand die aan de cursor hangt in Micro-Cap nog steeds de zigzag-vorm heeft, dien je Micro-Cap nog te sluiten en opnieuw op te starten.

De weerstand ziet er nu zo uit:



## 1.2. Schema tekenen

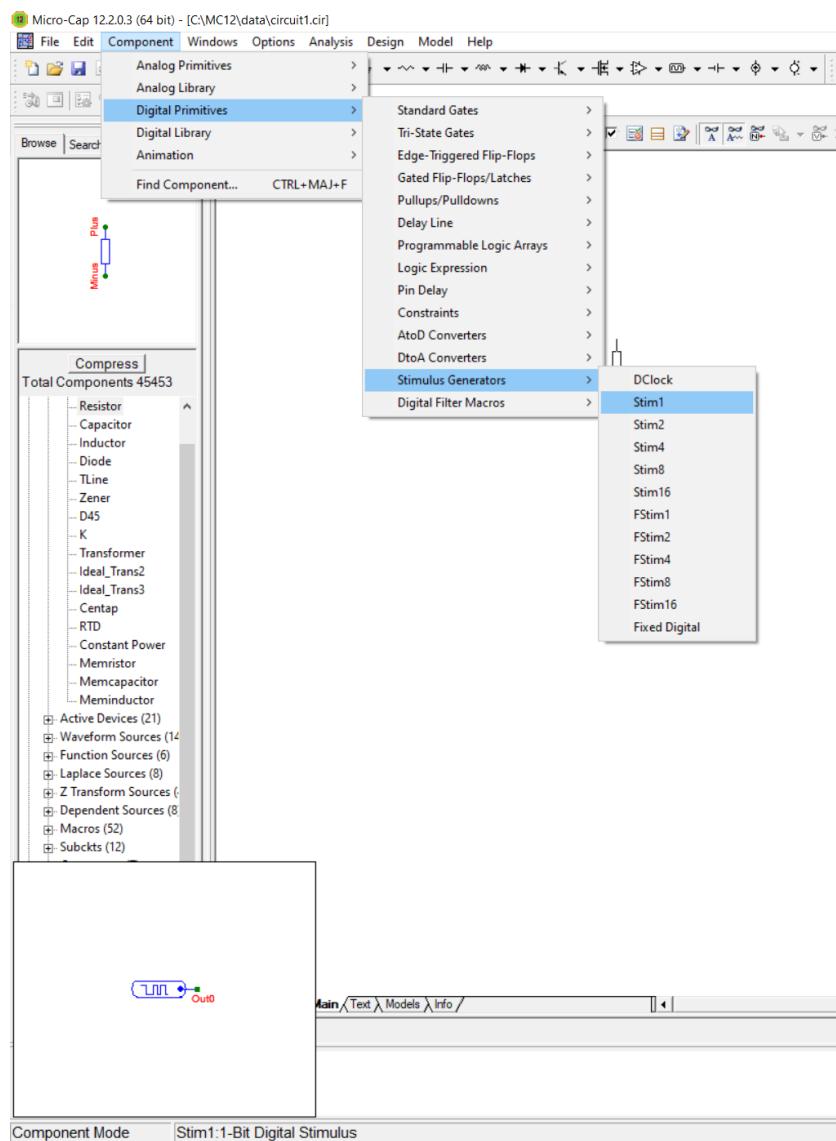
Om een schema te tekenen, ga je eerst de componenten plaatsen en dan de verbindingen leggen. Voor de eerste schakeling stuurt een digitale signaalgenerator een 1-bits signaal naar een NAND-poort die geschakeld is als een inverter.

Kies als signaalbron de **digitale signaalgenerator met een 1-bits uitgangssignaal**. Micro-Cap noemt dit **Stim1** en je kan dit vergelijken met de functiegenerator in het labo die je instelt als een blokgolf. Selecteer de component via het menu ofwel via het linker scherm. We bekijken de twee mogelijkheden.

### Ofwel via het menu:

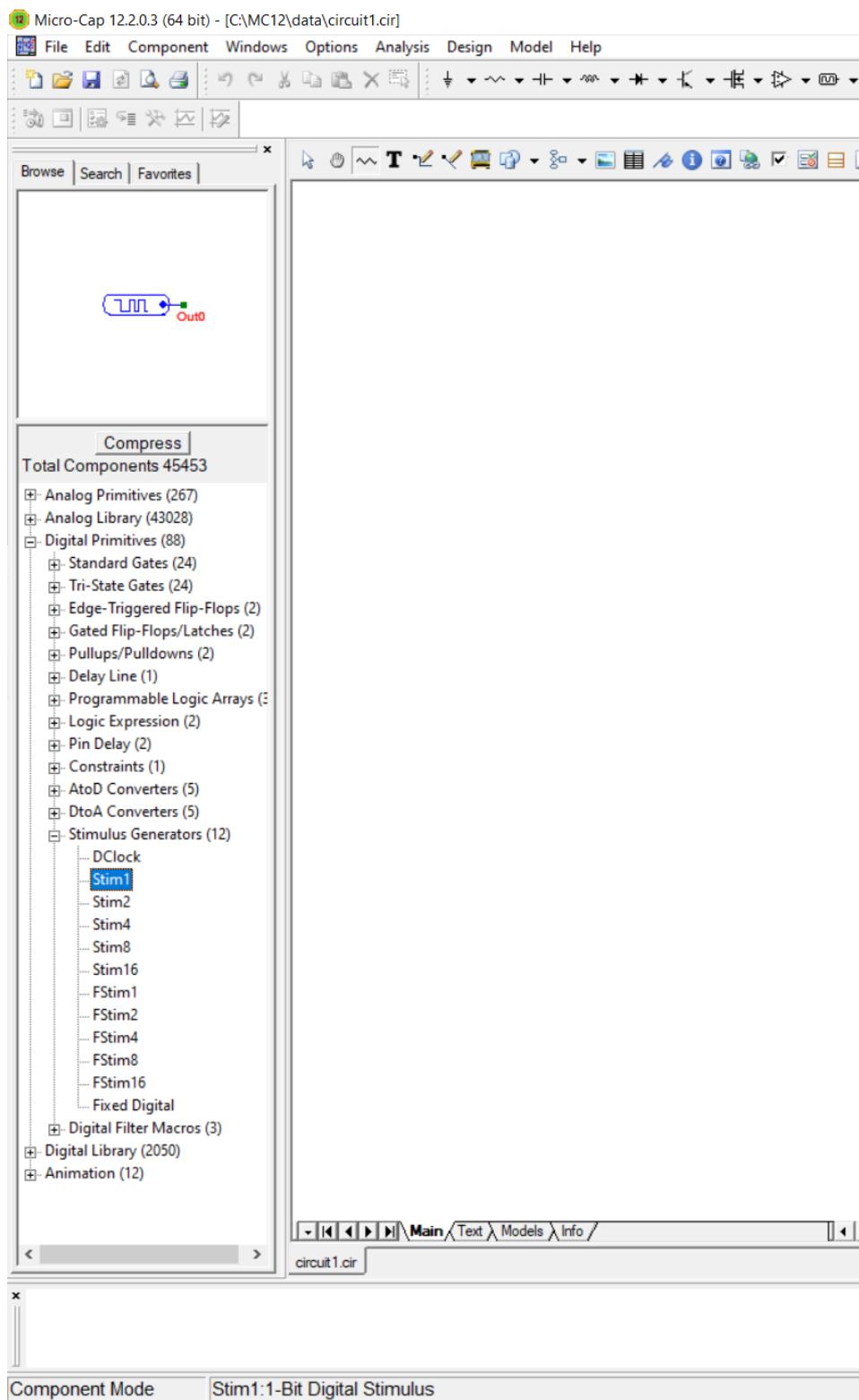
Klik in het menu op Component/Digital Primitives/Stimulus Generators/Stim1.

Je ziet linksonder uw scherm de symbolen verschijnen, nog voor je ze selecteert.



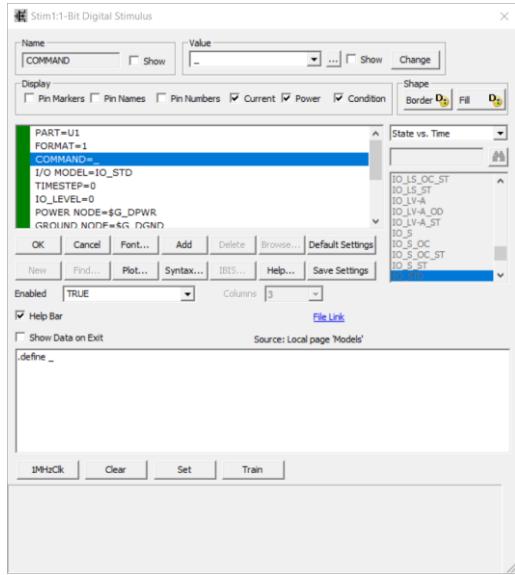
### Ofwel via het linker scherm:

Dit scherm noemt het *Panel*. Indien het *panel* niet zichtbaar is, kan je via het menu *Options/Panel* aanvinken of via de toetsencombinatie CTRL+ALT+X.

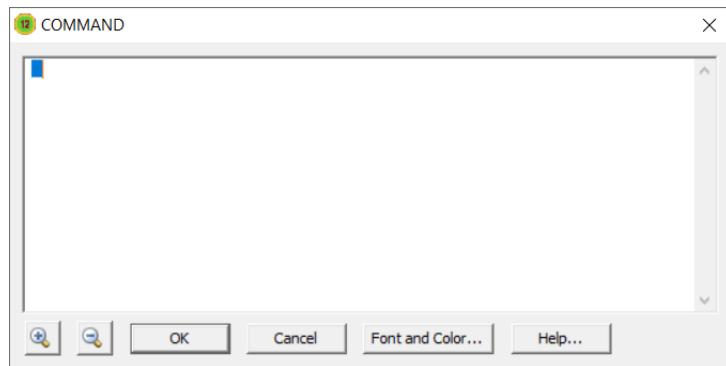


De geselecteerde component hangt nu aan je cursor. Beweeg je cursor naar een plaats links op het tekenblad en plaats de component door een druk op de linkermuisknop (LMK).

Er verschijnt een pop-up venster:

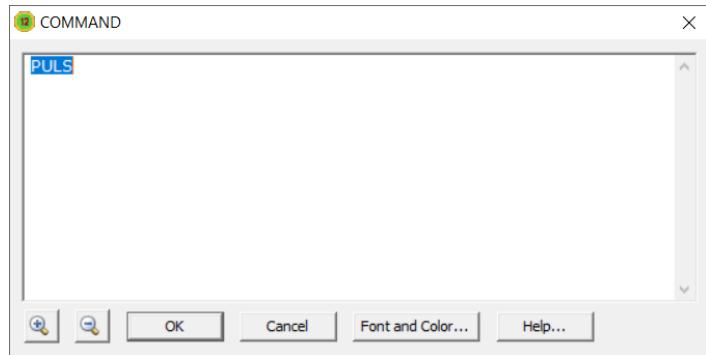


Hoe het digitale signaal er uit ziet (m.a.w. wanneer je een laag en hoog signaal wil), kan je ingeven bij **COMMAND**. Dubbelklik op de ‘COMMAND=’-attribute of klik bij *Value* op de knop. Een nieuw venster opent:



Geef nu een naam in (van de .define-variabele waar we straks de stimuluscommando's gaan aan toekennen). **Namen moeten starten met een letter of een underscore. De volgende karakters mogen letters, underscores of cijfers zijn.**

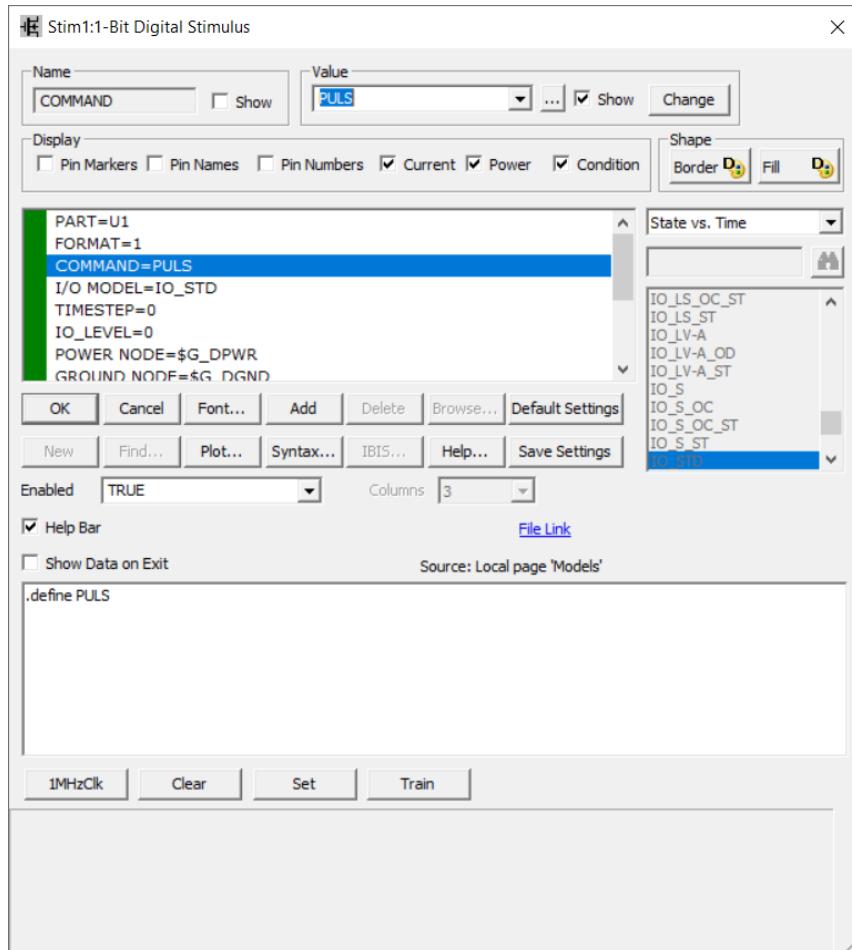
Verwijder eerst de underscore en kies als naam ‘**PULS**’ (opm. kleine letters worden straks automatisch omgezet naar hoofdletters, je kan beter steeds hoofdletters intypen).



Bevestig door op de knop OK te drukken.

**Vink steeds *Show* bovenaan rechts bij *Value* aan zodat de ingestelde naam straks ook zichtbaar wordt op het schema (dus niet de *Show* bij *Name*)!**

In het attribute-dialoogschermscherm (dit is het grootste scherm onderaan) is het .define-statement zichtbaar en kan aangevuld worden.



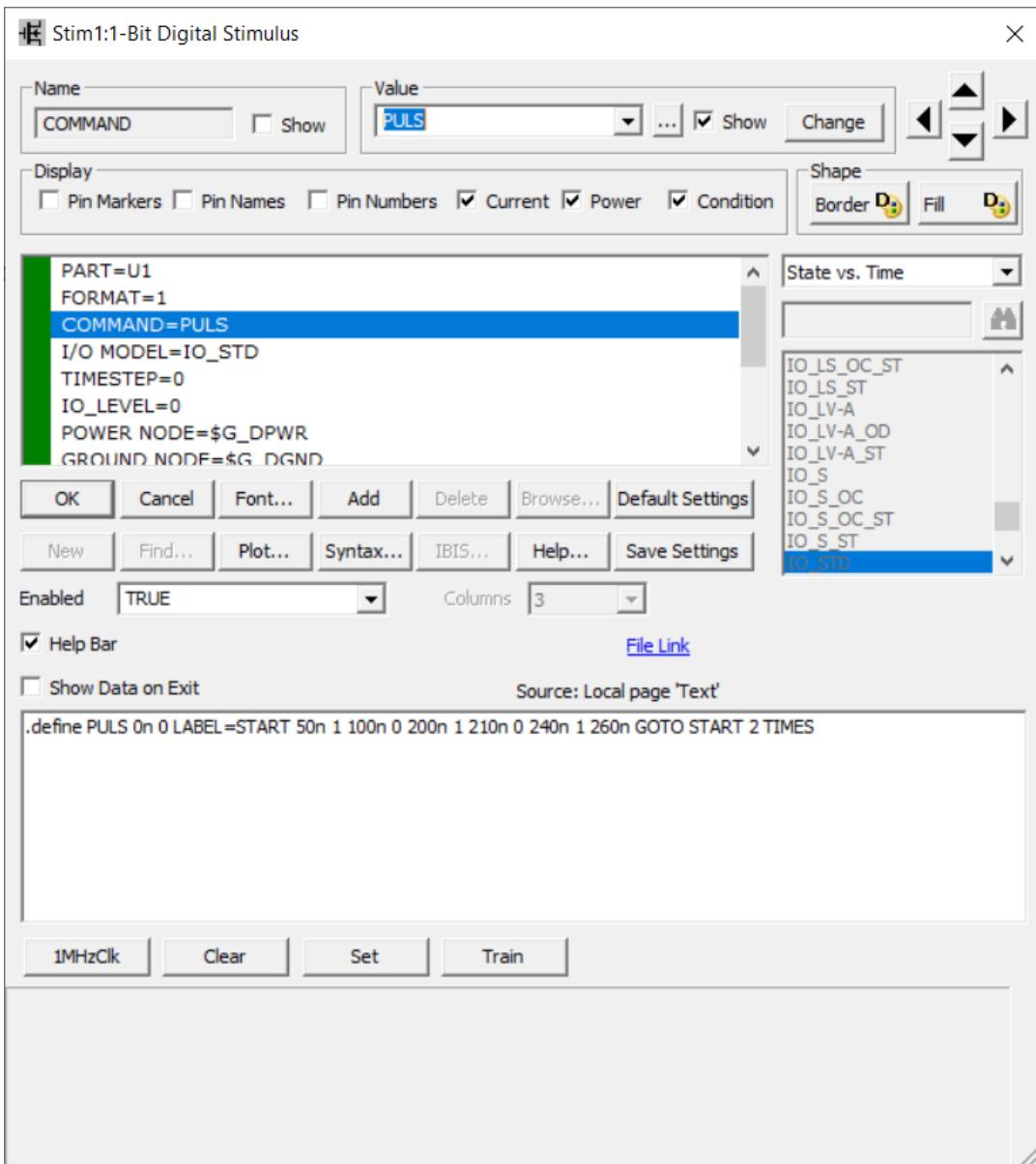
Geef in het attribute-dialoogscherm nu de gegevens in uit de eerste kolom van de onderstaande tabel. Zet deze gegevens na elkaar gescheiden door een spatie.

**Opgelet:**

1. typ een ‘nul’ en geen letter ‘o’!
2. Bemerk dat er geen spatie staat bij de tijd en wel een spatie tussen de tijd en het logische niveau 0 of 1.
3. Je mag zowel n (=nano) als ns (nanoseconde) noteren.

<b>Notatie</b>	<b>Betekenis</b>
0n 0	Op tijdstip 0 ns is het uitgangssignaal 0
LABEL=START	Plaats een label met naam START op dit punt in de tijdreeks
50n 1	50 ns (na tijdstip 0) wordt de uitgang hoog
100n 0	Na 100 ns (na tijdstip 0) wordt de uitgang laag
200n 1	Na 200 ns (na tijdstip 0) wordt de uitgang hoog
210n 0	Na 210 ns (na tijdstip 0) wordt de uitgang laag
240n 1	Na 240 ns (na tijdstip 0) wordt de uitgang hoog
260n GOTO START	Spring naar het label START en ga daar verder
2 TIMES	Doorloop de lus tweemaal

Het Stim1-venster zou er nu zo moeten uitzien:



Sluit het venster van STIM1 af door op OK te klikken (dit staat in het midden links van dit venster).

Opmerking:

1. Bestudeer de betekenis en de notatie in de tabel zodat je later zelf een andere puls kan ingeven!
2. Indien je voor een andere toepassing een oneindige lus zou willen, vervang je '2 Times' door '**-1 Times**'.

De stimulus-generator met zijn COMMAND-naam wordt op het schema geplaatst en ziet er zo uit:



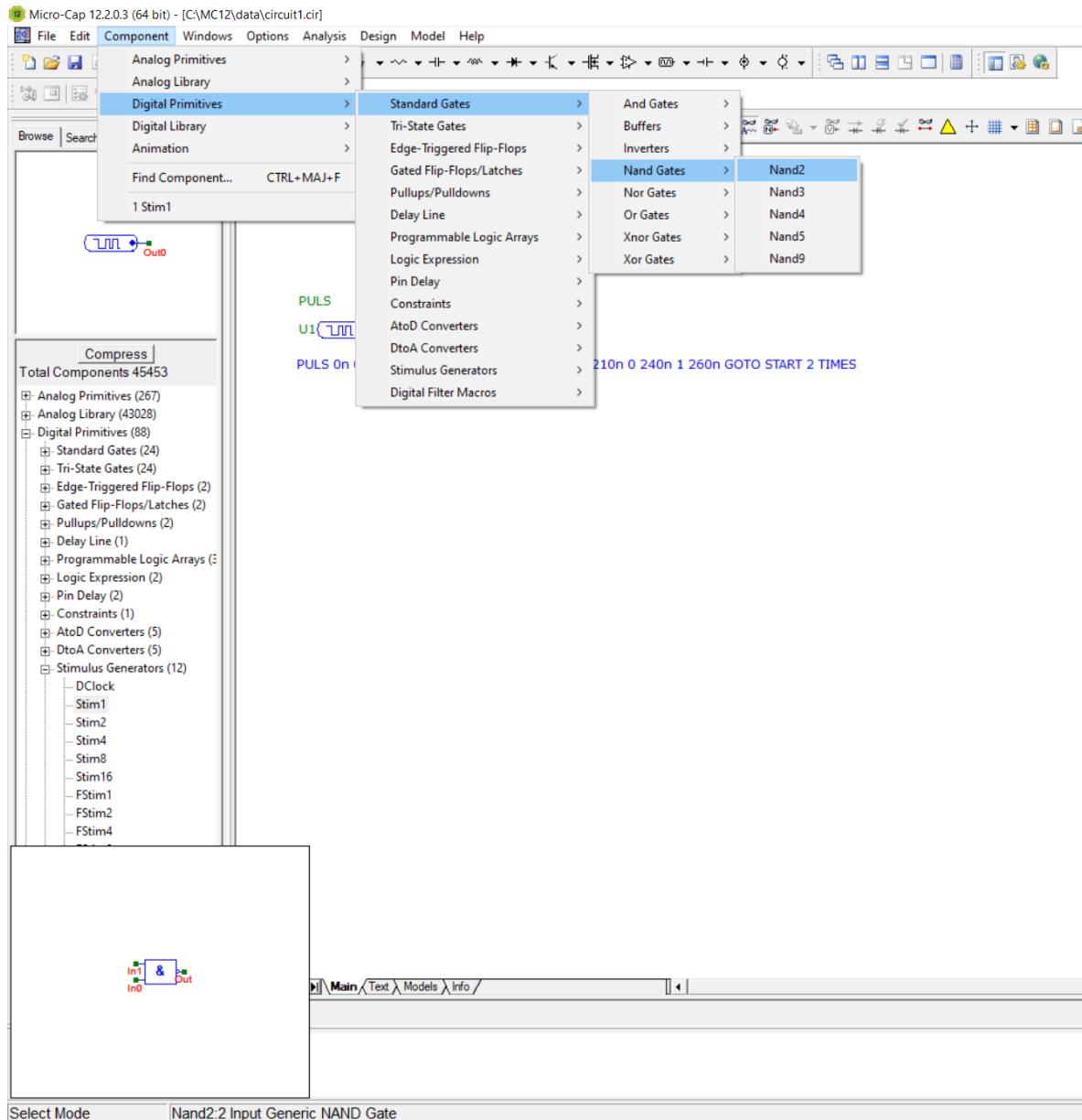
Het is belangrijk om **de instelling van de generator ook op het schema te zetten**, zodat je (of de docent(e) later bij het opsporen van eventuele fouten) de instelling op de afdruk kan zien. Dubbelklik opnieuw op het symbool van de stimulusgenerator en kopieer de tekst na .define (dus zonder “.define”) !

Plaats met een tekstbox onder de generator en plak er de geselecteerde tekst in. Je krijgt het volgende resultaat:

PULS  
U1  
PULS On 0 LABEL=START 50n 1 100n 0 200n 1 210n 0 240n 1 260n GOTO START 2 TIMES

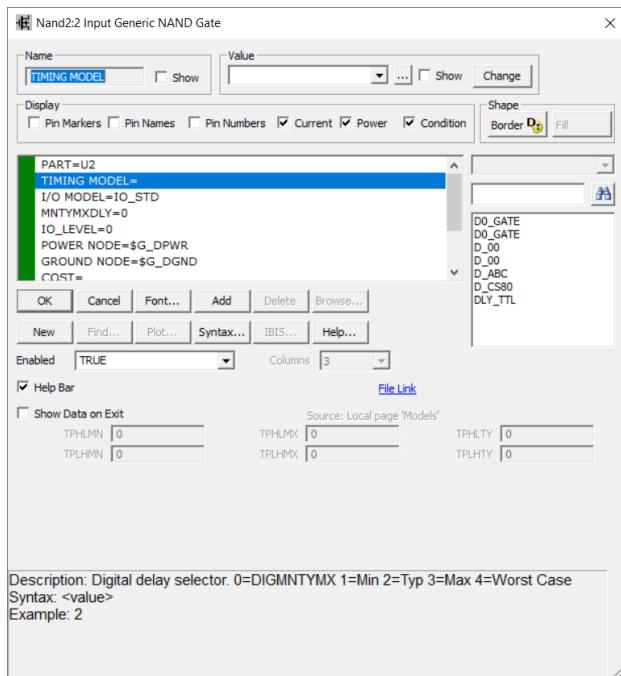
We vervolledigen het schema met een NEN-poort met 2 ingangen (deze heeft dezelfde functie van een EN-poort met een invertor).

De uitgang van de generator gaat naar een ‘**2 input NAND Gate**’. Ga op dezelfde wijze tewerk als hier voor om deze component op te halen. Dit kan opnieuw via het *panel* of via het menu.



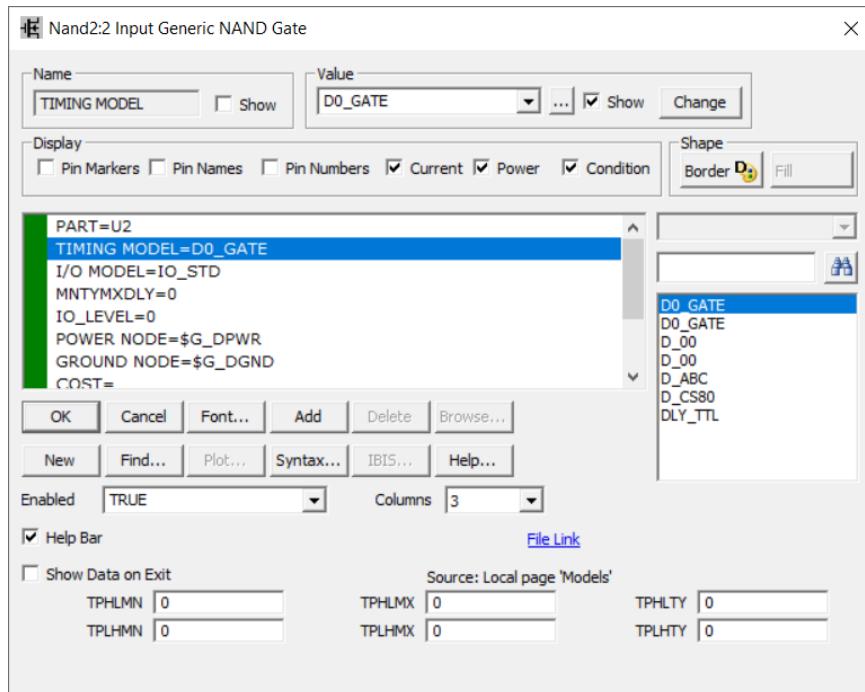
Controleer hierbij dat hetzelfde **IEC-symbool** aan je cursor hangt zoals je links onder op de bovenstaande figuur ziet!

Je kan dan de parameters voor de Nand2-poort aanpassen in het pop-up scherm:



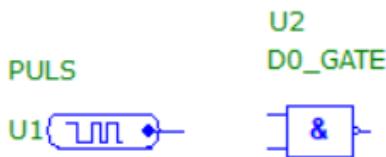
Het ‘**TIMING MODEL**’ wordt ingesteld op **D0\_GATE**. Dit geeft geen vertraging in de simulatie, m.a.w. de poorten beschouw je als ideaal. Je kan dit aanklikken rechts in het venster. (Een andere keuze is *DLY\_TTL* en geeft de standaard vertraging tussen de in- en uitgang voor de TTL-familie.)

Terug Show aanvinken bij Value zodat de ‘*timing model*’-instelling zichtbaar wordt.





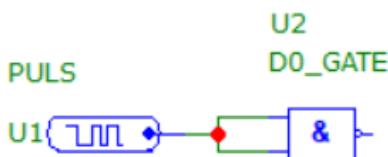
Via de **Select Mode**-pijl (CTRL+E) kan je componenten of eventueel overlappende teksten verplaatsen (klik op tekst, houd de LMK ingedrukt en versleep). Je bekomt nu zoiets:



```
PULS On 0 LABEL=START 50n 1 100n 0 200n 1 210n 0 240n 1 260n GOTO START 2 TIMES
```



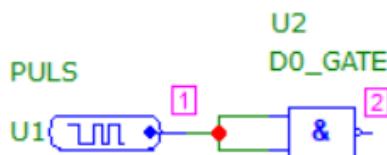
Via de **Wire Mode** (CTRL+W) leg je de verbindingen.



```
PULS On 0 LABEL=START 50n 1 100n 0 200n 1 210n 0 240n 1 260n GOTO START 2 TIMES
```



Via de **Node Numbers** wordt het nummer van het knooppunt zichtbaar.



```
PULS On 0 LABEL=START 50n 1 100n 0 200n 1 210n 0 240n 1 260n GOTO START 2 TIMES
```

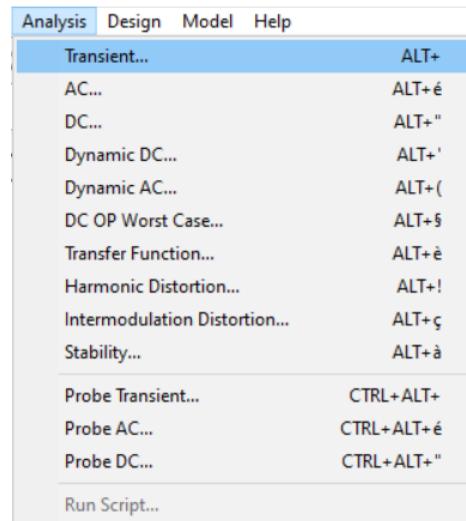
**Fig. 1: Eenvoudig beginnen: een digitale signaalgenerator stuurt een 1-bits signaal naar een inverter**

**Verplaats steeds de *node numbers* als ze het invertor-teken (bolletje) van de uitgang van een poort bedekken!** Het is immers belangrijk dat je in het schema ziet of het een AND-poort of een NAND-poort is. Via de *Select Mode*-pijl (selecteren en ingedrukt houden) zijn de nummers van de nodes (knooppunten) verplaatsbaar zodat de verbindingen volledig te zien zijn.

Proficiat, je eerste schema is getekend! Nu kan je hiermee een simulatie uitvoeren. Dit is een snelle manier om schakelingen te ontwikkelen; je hebt immers geen fysische componenten nodig.

### 1.3. Simulatie

Voor de **analyse** van digitale schakelingen wordt gekozen voor **Transiënt** (= overgangsverschijnsel) **analyse** omdat deze meest relevant is voor digitale schakelingen (pulsdiagram) terwijl AC en DC analyse meer bij analoge schakelingen gebruikt wordt.



Bij de instellingen in het ‘*Transient Analysis Limits*’-venster dat werd geopend, wordt de *Maximum Run Time* aangepast naar 700ns zodat de tijdslus zichtbaar wordt gemaakt, die tweemaal doorlopen wordt. Controleer dat *Auto Scale Ranges* is aangevinkt zodat de pulsdiagrammen automatisch geschaald worden. De ‘*X Expression*’ is de tijd T op de x-as en de ‘*Y Expression*’ stelt het digitale niveau d(1) en d(2) van de respectievelijke node 1 en 2 van het schema voor op de y-as.

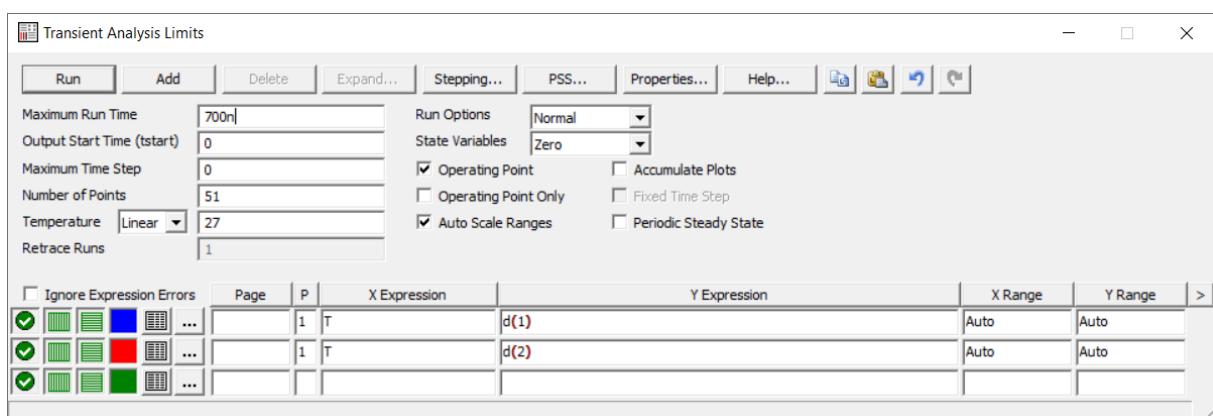


Fig. 2: Het transiënt-analyse venster

Kies nu voor **Run** linksboven.

Het resultaat is:

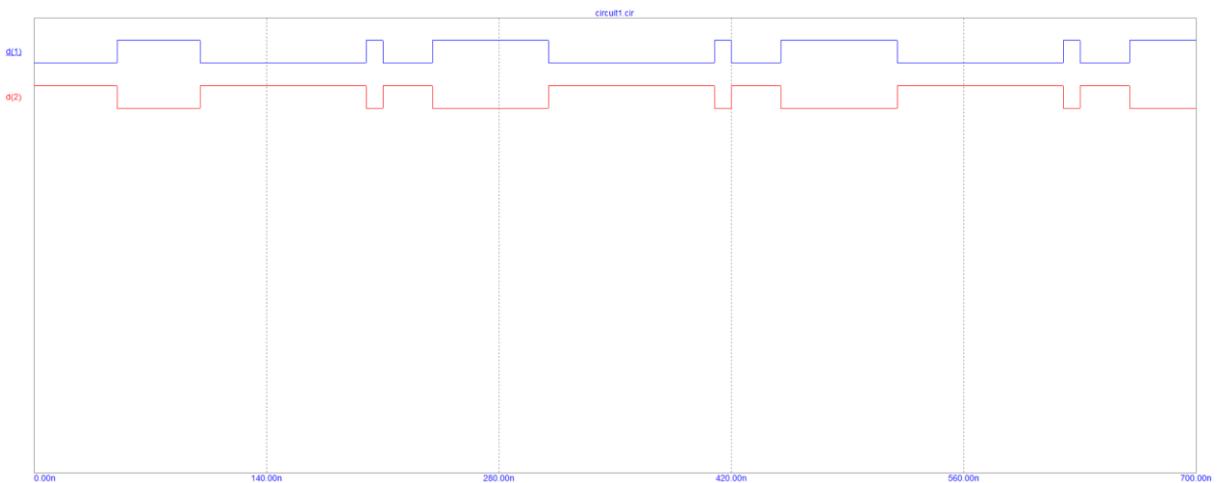
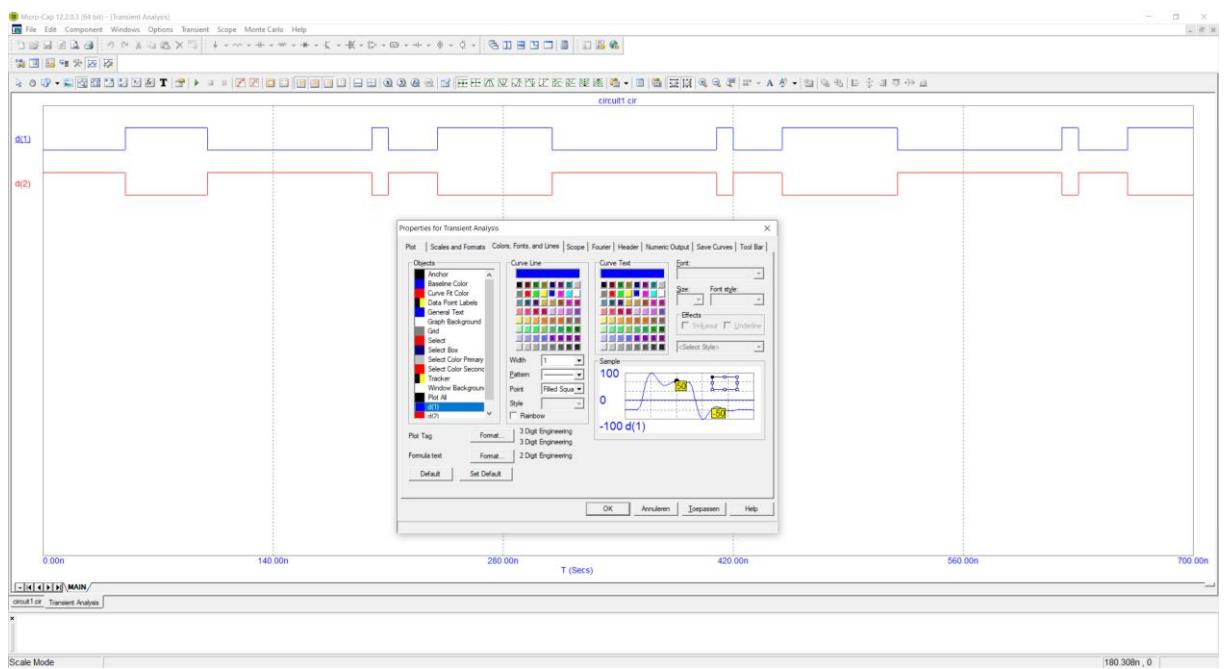


Fig. 3: Dit zijn de resulterende golfvormen bij een transiënt-analyse die is uitgevoerd op de schakeling in Fig. 1

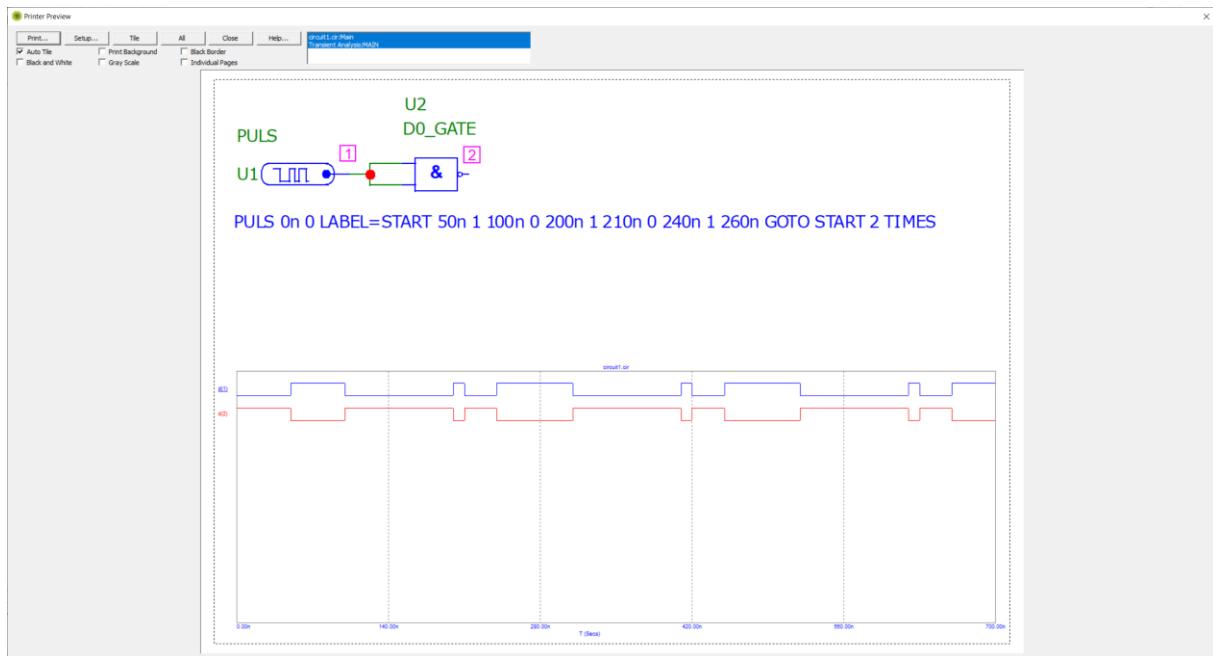
Via de knop **Properties...** uit figuur 2 (of F10) en de bovencommandolijn kan je een pak zaken aanpassen. Verken deze zaken eens op vrije basis:



## 1.4. Afdrukken

**Druk uw schema en simulatie af in kleur op papier.** Omdat het schema beperkt is in grootte, kan je ook de simulatie op één blad mee afdrukken.

- Ga via File/Print Preview... en vink ook ‘Auto Tile’ aan.
- Selecteer beide bestanden die rechtsboven worden weergegeven (SHIFT + bestand klikken). Beide bestanden kleuren blauw.



Druk op de knop ‘Print...’

Je kan nu je printer selecteren. Druk OK om het afdrukken te starten.

## 1.5. Opslaan

Sla uw schakeling op via File\Save As...\

Geef het bestand de naam “Familienaam\_Voornaam\_Inleiding schakeling 1”.

De standaard map is c:\MC12\DATA\ . Je mag deze map wijzigen. Sla steeds je bestanden op zodat je ze kan benaderen zowel van thuis als vanop de hogeschool.

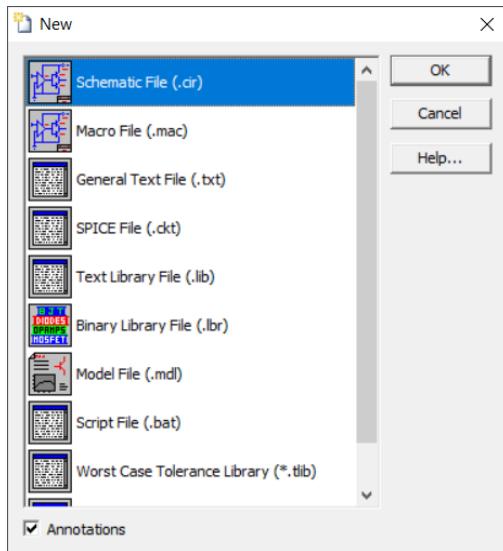
## 1.6. Schakeling afsluiten

Sluit deze schakeling door op File → Close te drukken.

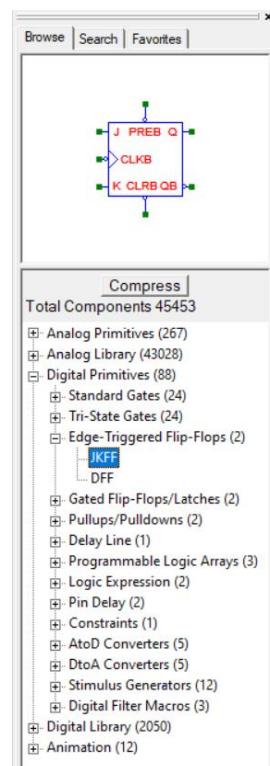
## 1.7. Pulsteller

Wij gaan nu een nieuwe uitgebreidere schakeling uittesten, namelijk een pulsteller. De hiervoor geziene werkwijzen blijven we toepassen.

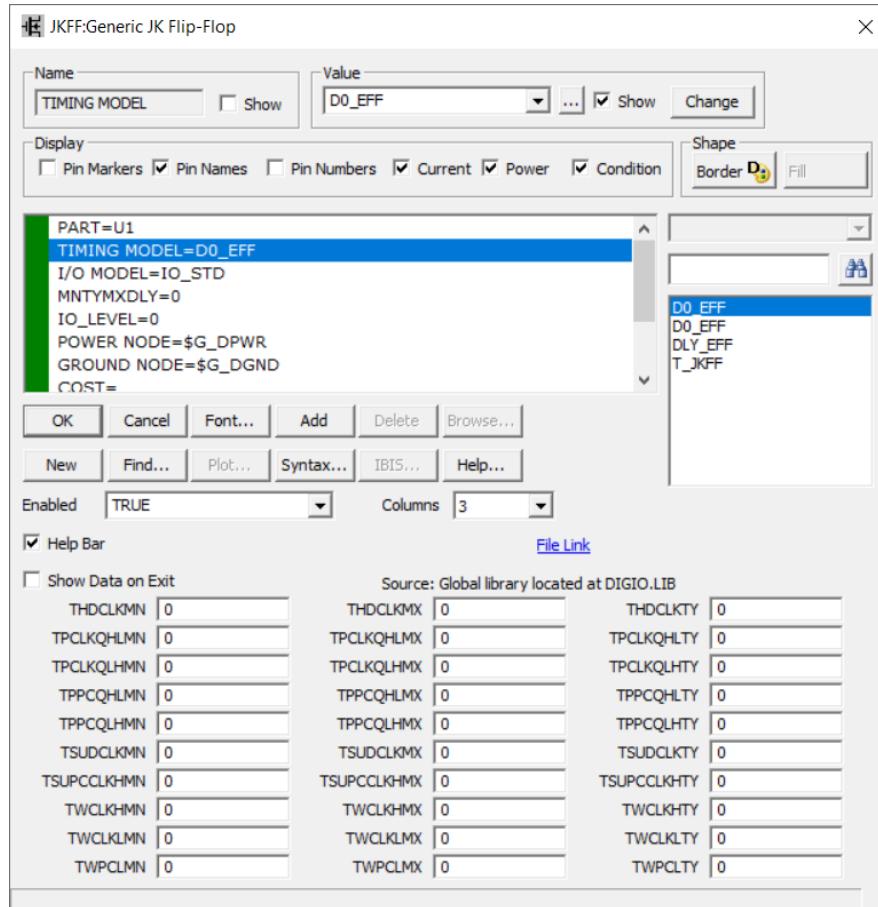
Start een nieuw tekenblad via File → New... → Schematic File (.cir)



De pulsteller wordt opgebouwd uit drie edge triggered JK flipflops (de eigenlijke werking leren we later nog). We roepen de eerste component op en bemerken dat deze afgekort wordt als JKFF. Plaats deze component op het blad.



We kiezen terug voor het timing model zonder vertraging **D0\_EFF** wat we ook zichtbaar maken door Show aan te vinken.



De drie JKFF's worden aangestuurd door drie 1-bits stimulus generatoren (zie instellingen op het schema hieronder). Werk het schema af naar het voorbeeld in figuur 4. De nodes kunnen een verschillend nummer krijgen door een andere plaatsingsvolgorde van de componenten – dit is normaal. Verplaats de nodes zodat alle aansluitingen van de componenten goed zichtbaar zijn.

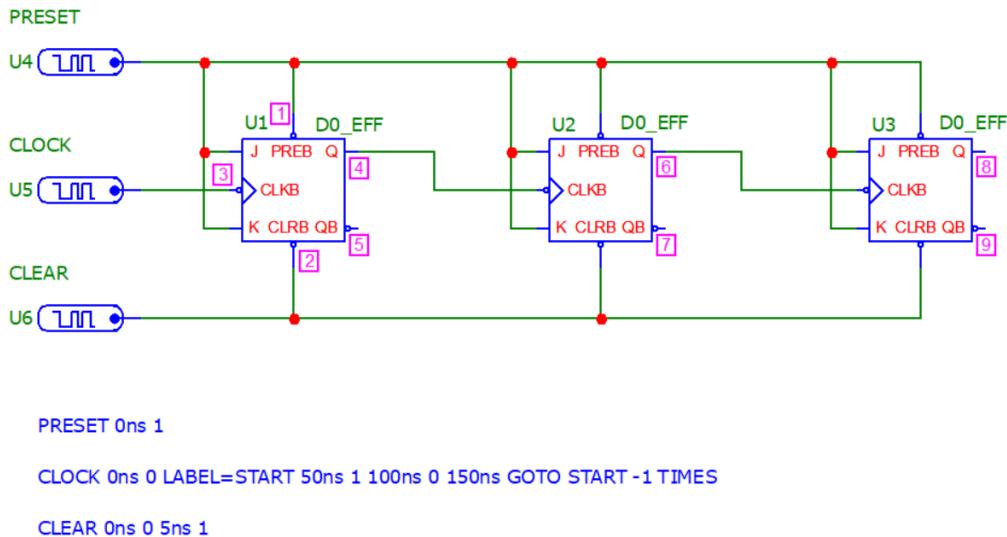


Fig. 4: Een driebits teller die is opgebouwd met drie JK-flipflops

Verklaring command-instelling stimulus generatoren (zie tabel 2):

**Tabel 2**

Generator	Notatie	Betekenis
U4	COMMAND = PRESET 0ns 1	Constant hoog niveau op J, K en SET van alle flipflops
U5	COMMAND = CLOCK 0ns 0 LABEL=START 50ns 1 100ns 0 150ns GOTO START -1 TIMES	De klok produceert 50 ns-pulsen ( $f = 10 \text{ MHz}$ )
U6	COMMAND = CLEAR 0ns 0 5ns 1	Geeft een korte laag-puls om de flipflop te resetten

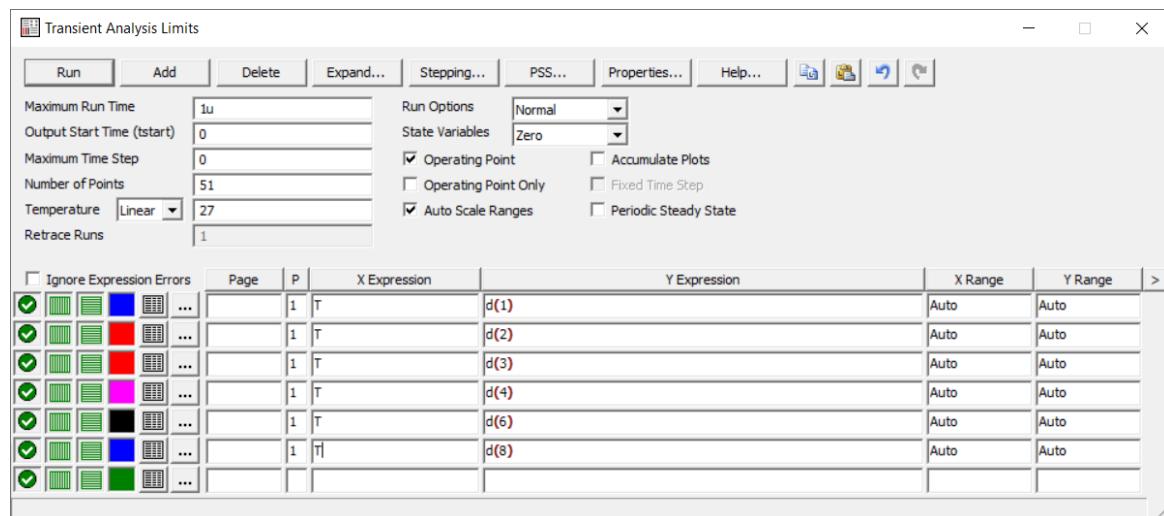
In het symbool voor de JK-flipflop zien we de afkortingen:

- PREB staat voor PresetBar wat eigenlijk een actief lage ingang is. De ‘Bar’ in Micro-Cap is de letteraanduiding voor de overstreping of negatie. In sommige datasheets kan dit ook vermeld staan als  $\overline{\text{PRESET}}$  of  $\overline{\text{SET}}$  (zie ook de theorie).
- CLRB staat voor ClearBar. In datasheets kan dit vermeld worden als  $\overline{\text{CLEAR}}$ .

Zet steeds de nodenummers aan!

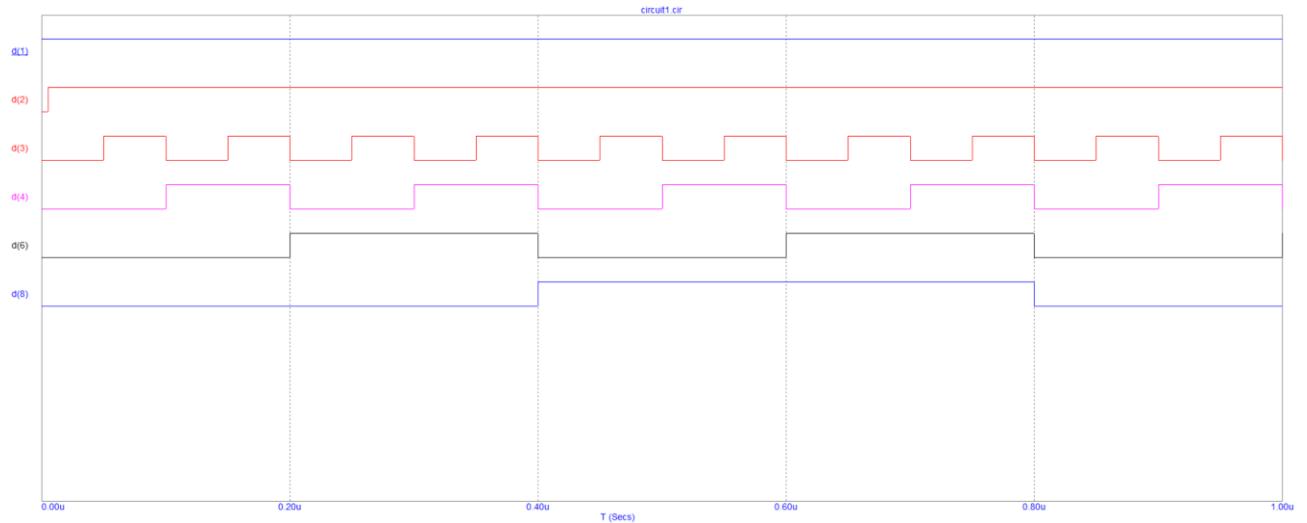
Voeg via  de volgende knooppunten toe aan de transiëntanalyse: 3,4, 6 en 8. Dan zie je straks alle signalen van de generatoren eerst staan gevolgd door de drie Q-uitgangen van de flipflops. Vergeet dus niet om de waarde in d( ) aan te passen. Eventueel kan je de kleuren nog aanpassen.

*Maximum Run Time* is  $1\mu\text{s}$  om de volledige telcyclus te zien. Je geeft dit in als ‘1u’.



Indien de nodenummers in uw schema anders toegekend werden, dan pas je de overeenkomstige nodenummers aan.

Resultaat:



**Fig. 5: Dit zijn de signaalvormen die ontstaan als we een transiënt-analyse uitvoeren op de teller**

- d(1) is continu hoog, zoals we hebben ingesteld met U4.
- d(2) geeft een korte 0-puls voor het resetten van de flipflops en wordt daarna hoog.
- d(3) toont het 10 MHz-kloksignaal. Controleer in de simulatie of dit klopt.
- d(4), d(6) en d(8) zijn de uitgangssignalen van de drie flipflops, waarbij goed te zien is dat elke flipflop het signaal van de voorgaande flipflop door twee deelt.

Zo telt deze teller continu van 000, 001, 010, 011, ... tot 111. Dit kan je makkelijk zelf controleren door op de kleurenafdruk van je simulatie voor elke klokpuls van d(3) bij de uitgangen d(4), d(6) en d(8) voor een laag niveau een nul en voor een hoog niveau een één te schrijven.

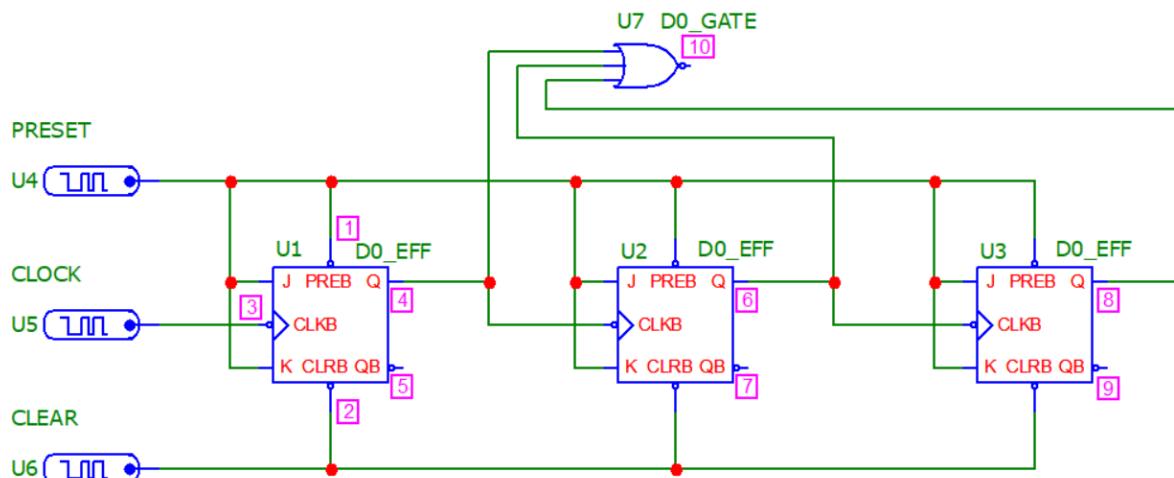
Controleer steeds de waarde rechts op de tijdas; deze geeft de ingestelde 1,00  $\mu$ s weer. Deze controle zal u helpen om fouten snel op te sporen.

Analyseer zelf ook nog eens de signalen op de inverterende uitgangen d(5), d(7) en d(9).

Druk de schakeling en simulatie af in kleur (zie richtlijnen vroeger) en sla de schakeling op als “Familienaam\_Voornaam\_Inleiding schakeling 2”.

## 1.8. Meer logica

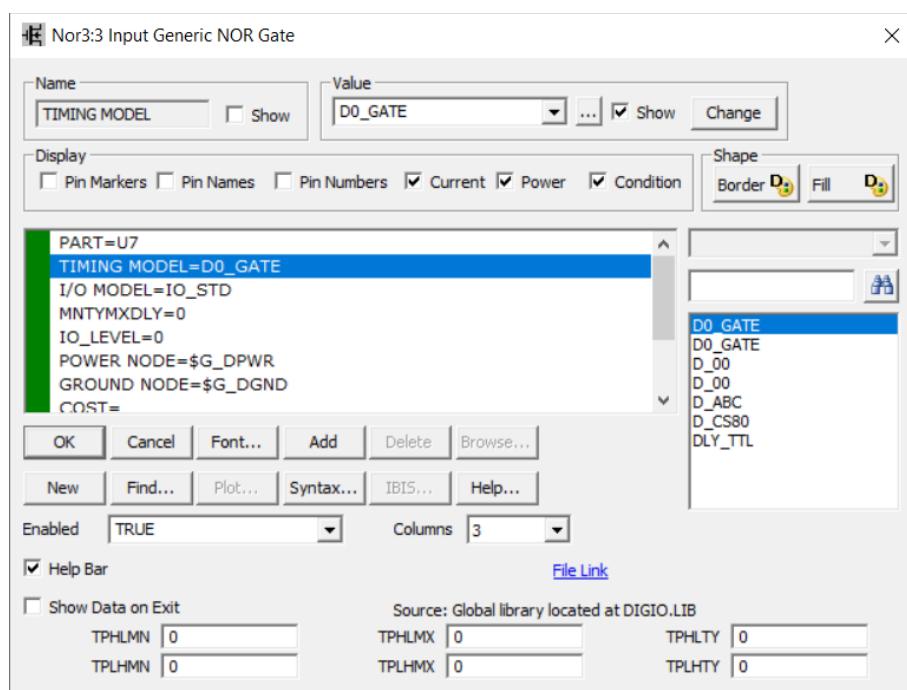
De verschillende tellerstanden kunnen met wat logica eenvoudig gedetecteerd worden. Het eenvoudigste is natuurlijk de stand 000. Dat kan met een NOF-poort met 3 ingangen; als alle drie ingangen laag zijn, wordt de uitgang hoog. Voeg zo'n NOF-poort toe aan de teller uit Fig. 4 en kies voor het timing-model *D0-gate*. Bemerk het bolletje als inversieteken aan de uitgang van de poort!



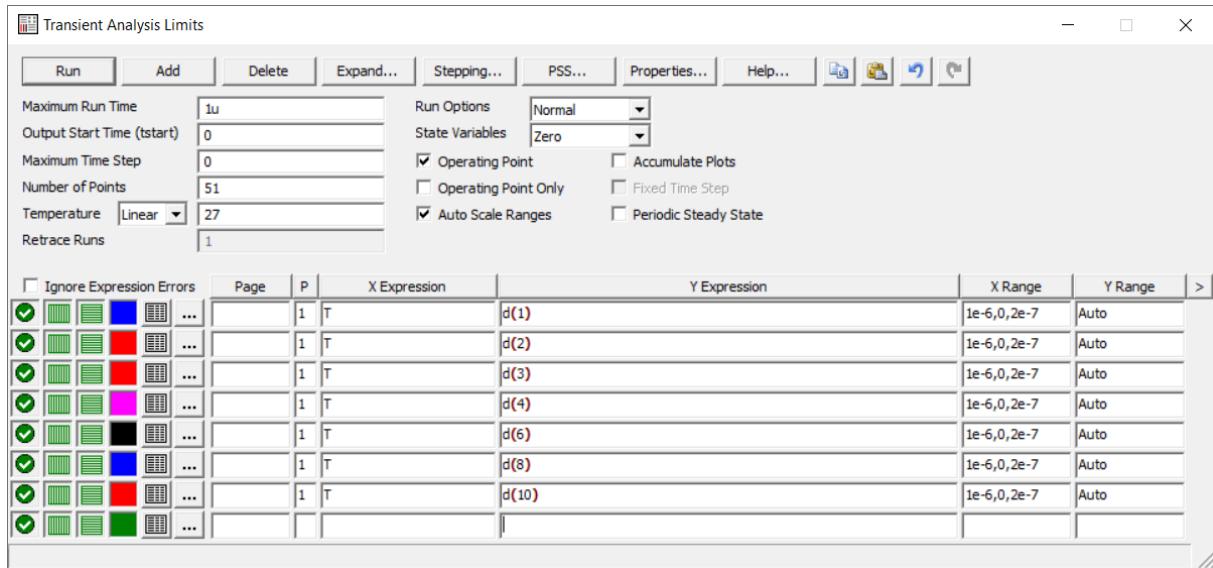
PRESET Ons 1

CLOCK Ons 0 LABEL=START 50ns 1 100ns 0 150ns GOTO START -1 TIMES

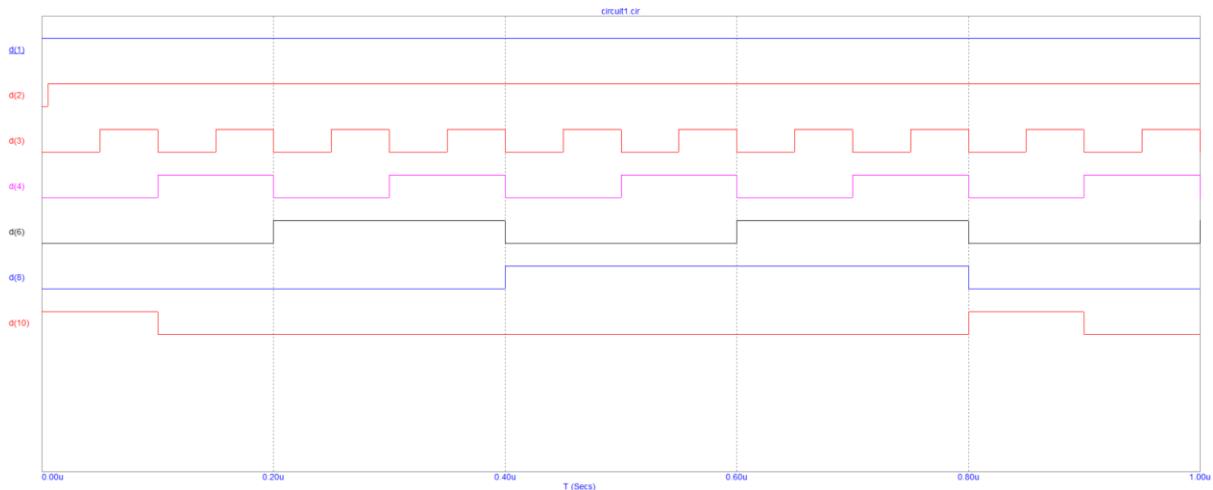
CLEAR Ons 0 5ns 1



We voegen uitgang 10 van de NOF-poort toe aan de transiënt-analyse:



In Figuur 6 is te zien dat de uitgang van de poort d(10) alleen hoog wordt als alle flipflop-uitgangen laag zijn.



**Fig. 6: Invloed van de NOF-poort met drie ingangen**

We zouden deze schakeling kunnen gebruiken om een LED af en toe te laten oplichten. Alleen moet dan een lagere frequentie dan de nu ingestelde 10 MHz gekozen worden, bijvoorbeeld 2 Hz. Dit is nodig voor de traagheid van het oog.

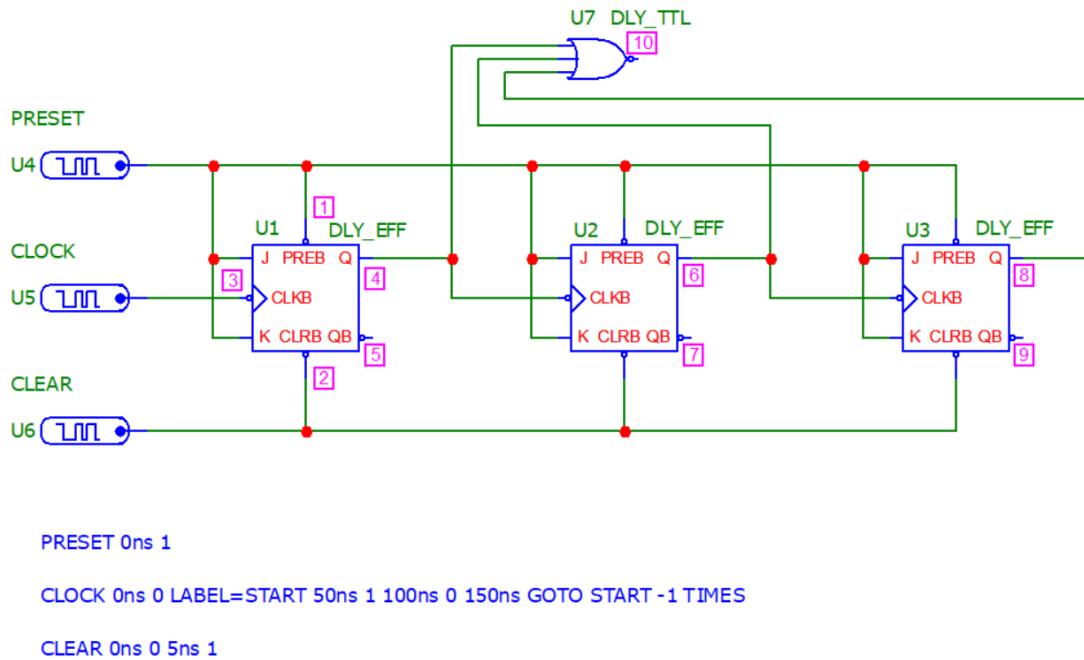
Druk de schakeling en simulatie af in kleur (zie richtlijnen vroeger). Staan de nodenummers op de juiste plaats?

Sla de schakeling op als “Familienaam\_Voornaam\_Inleiding schakeling 3”.

## 1.9. Vertraging

Bij schakelingen die met hoge frequenties werken, duikt een ander probleem op dat we gelukkig kunnen meenemen in de simulatie: de vertraging in de logische elementen. **Tot nu toe hebben we die vertraging nul gesteld (D0\_FF en D0\_GATE)**, zodat de resultaten netjes overeen komen met de bijbehorende waarheidstabellen. Een echte poort of flipflop verandert zijn uitgangstoestand niet onmiddellijk. Na een niveauverandering op een ingang duurt het een korte tijd voordat die verandering intern verwerkt is en effect heeft op een uitgang. **Zo is de typische vertragingstijd (propagation delay) van een TTL-poort ongeveer 11 ns.** We kunnen deze vertragingen inbouwen met behulp van de timing-modellen. Selecteer de muispijl linksboven en dubbelklik daarmee op een component. Het componentvenster verschijnt dan.

Kies bij de flipflops als timing-model **DLY\_EFF** en bij de NOR-poort **DLY\_TTL**.



Laat de transiënt-analyse opnieuw lopen en het resultaat ziet eruit zoals in Fig. 7.

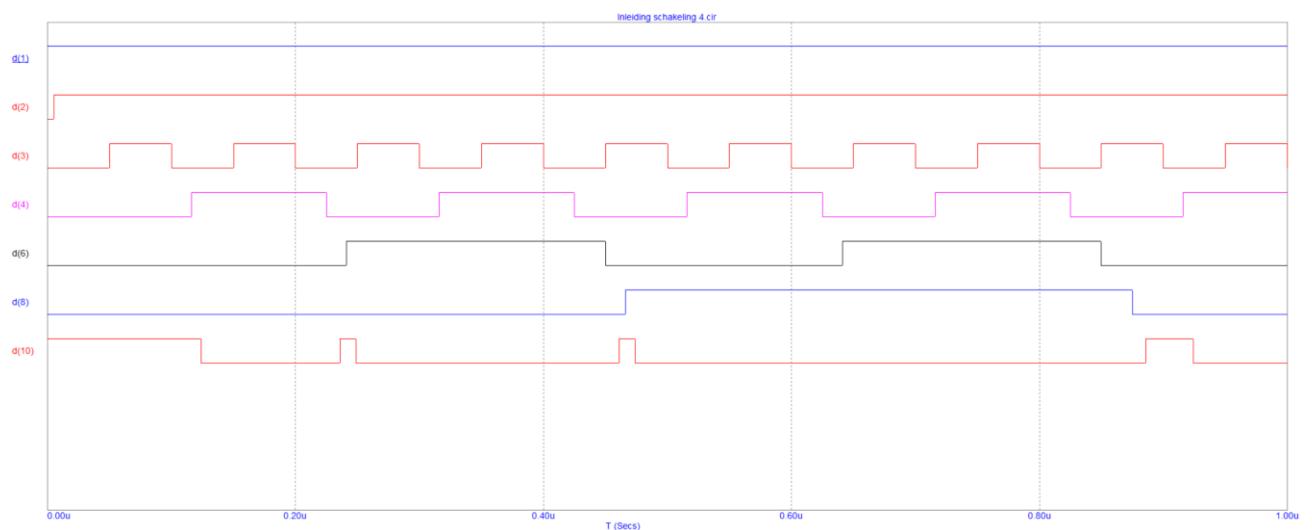


Fig. 7: Dezelfde schakeling, maar nu zijn modellen gebruikt met interne tijdsvertragingen

Duidelijk is nu de vertraging te zien die bij de verschillende signalen ontstaat. Het duurt altijd een korte tijd voordat de uitgang van de volgende flipflop reageert op een niveauwisseling op zijn ingang. Bij d(10) verschijnen echter enkele **ongewilde pulsen**, zogenaamde **glitches**.

Bij de tellerverandering van 001 naar 010 wordt d(6) pas hoog als d(4) al een aantal nanoseconden lang naar 0 gedaald is. Denk eraan dat d(4) het LSB (= Least Significant Bit = minst beduidende bit) is en dit rechts wordt genoteerd in het bitpatroon. Dit wordt door de NOF-poort gedetecteerd en als een korte puls aan zijn uitgang weergegeven. Het volgende moment waarop het mis gaat, is bij de wisseling van 011 naar 100. Dit verloopt via twee overgangsstappen, 010 en 000. Laatstgenoemde stand veroorzaakt weer een glitch (= een ongewenst overgangsverschijnsel). Tenslotte zien we nog dat bij de wisseling van 111 naar 000 de ‘1’-tijd aan de uitgang van de poort behoorlijk verkort wordt door de aanwezige vertragingstijden.

Bij langere tellercircuits kan het door de aanwezige vertragingen zelfs voorkomen dat de NOR-poort de simultane nulstand van alle uitgangen helemaal niet detecteert en dus geen ‘1’-puls produceert bij deze situatie.

Het timingsdiagram in Fig. 7 demonstreert dit serieuze probleem dat vooral bij hogesnelheidsschakelingen een grote rol speelt. Het toont ook het belang van simulatieprogramma’s aan, omdat daarmee al voor de bouw van de schakeling zulke problemen bestudeerd en opgelost kunnen worden. **Zulke glitches ontstaan weliswaar ook bij schakelingen die met veel lagere frequenties werken**, maar daarbij zijn de stoorpulsen t.o.v. de overige pulsen zo extreem kort dat ze geen invloed op de werking uitoefenen. In zo’n geval kan men de vertragingsnabootsing in digitale componenten rustig ‘uit’ zetten. **Bij hogere snelheden, zoals bij de hier getoonde teller, is het probleem zo erg dat de teller anders ontworpen moet worden** (bijvoorbeeld een synchrone teller gebruiken i.p.v. de afgebeelde ripple-counter).

Druk de schakeling en simulatie af in kleur (zie richtlijnen vroeger). Staan de nodenummers op de juiste plaats?

Sla de schakeling op als “Familienaam\_Voornaam\_Inleiding schakeling 4” en sluit dit schema af.

## 1.10. Initialisatie van de flipflops bij opstarten

We starten met een nieuw digitaal circuit dat bestaat uit een ringteller (Fig. 8) waarvan steeds één van de uitgangen beurtelings hoog wordt. De teller is opgebouwd met vier D-flipflops.

Selecteer D0\_EFF als timing-model voor de flank-getriggerde flipflop (**DFF**). Er wordt gebruik gemaakt van twee 1-bits generatoren (Stim1), waarvan de instellingen in tabel 3 gegeven zijn.

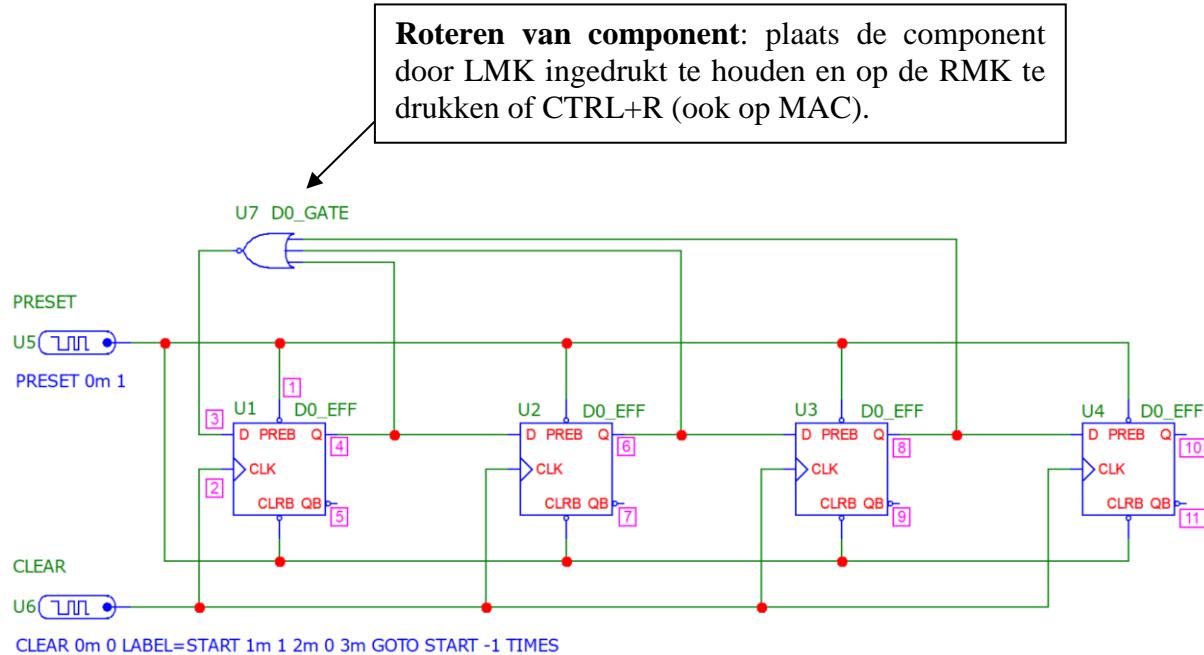


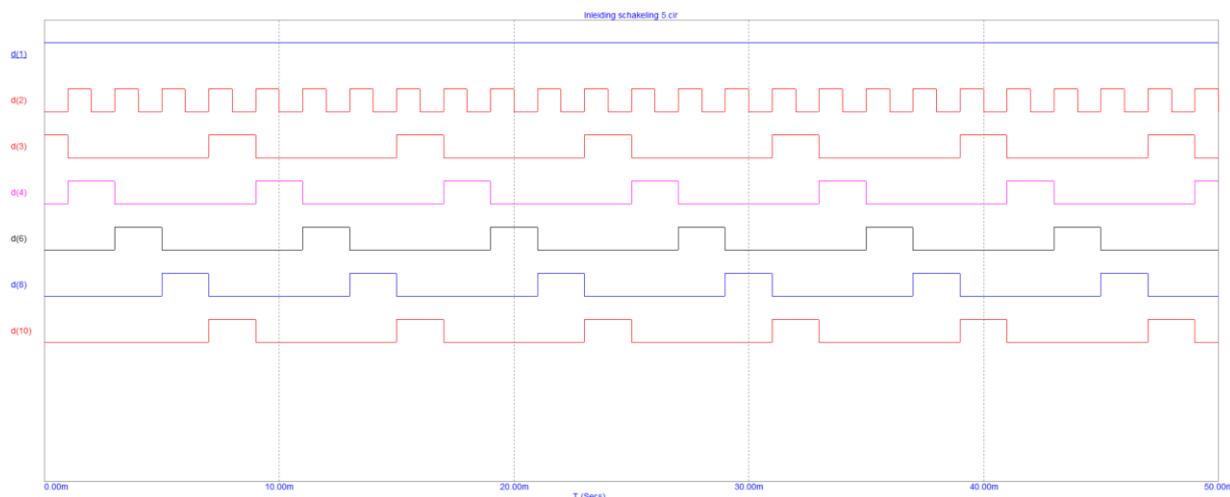
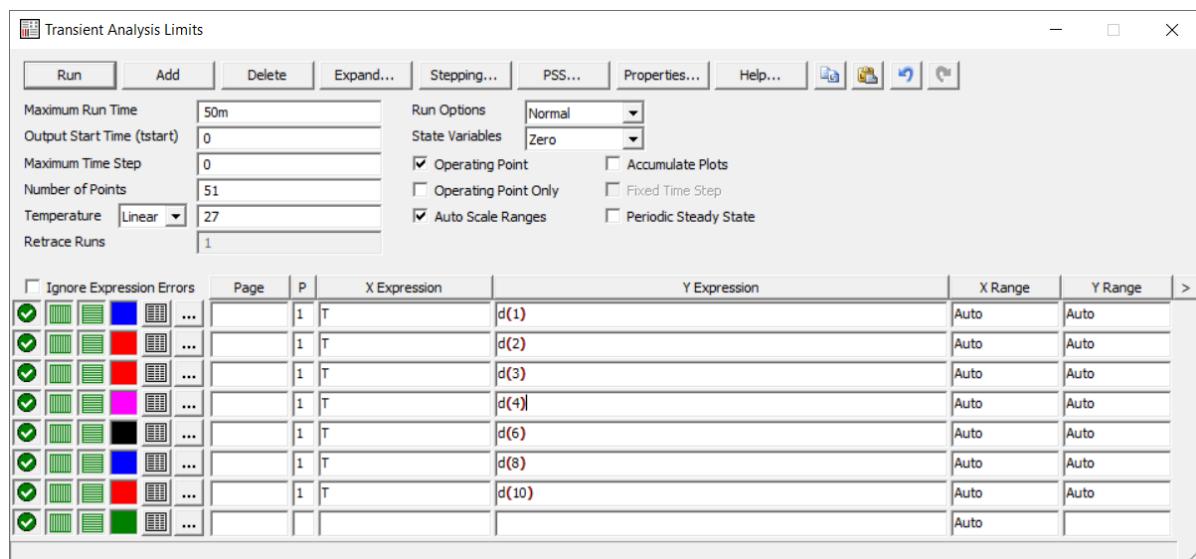
Fig. 8: Het schema van een ringteller

Tabel 3

Generator	Notatie	Betekenis
U5	COMMAND = PRESET 0m 1	Houd de set- en reset-ingang hoog
U6	COMMAND = CLEAR 0m 0 LABEL=START 1m 1 2m 0 3m GOTO START -1 TIMES	500 Hz klok

Indien de nummering U1, U2, ... van de componenten verschillend is, heeft dit te maken met de volgorde waarmee de componenten geplaatst zijn. De toekenning van de nummering verloopt immers automatisch. Bij een verschillende nummering dien je straks bij de simulatie de overeenkomstige nodes aan te passen.

Een NOF-poort met drie ingangen (timing-model D0\_GATE) koppelt de uitgangen van de eerste drie flipflops terug naar de D-ingang van de eerste flipflop. Simuleer de schakeling nu eerst.



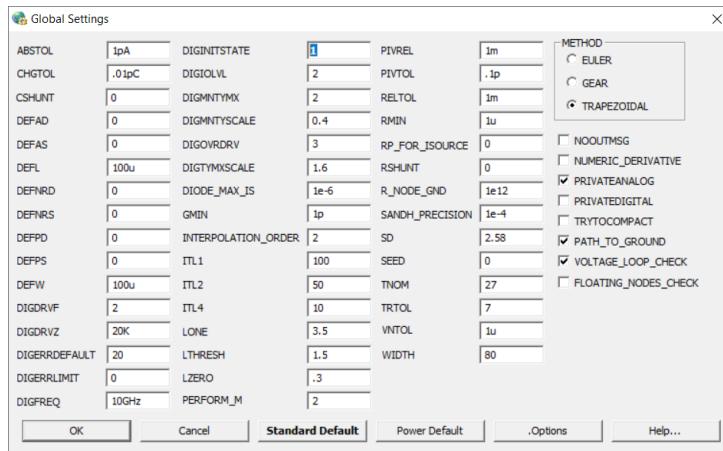
Bij het opstarten van de schakeling wordt de uitgang van de NOF-poort hoog want zijn ingangen zijn immers laag. Dit hoge signaal is aangesloten op de D-ingang van de eerste flipflop. Bij **de stijgende flank van de eerste klokpuls** wordt het hoge niveau aan de ingang van de eerste flipflop doorgegeven naar de uitgang van de eerste flipflop. Dit hoge niveau is verbonden met de D-ingang van de tweede flipflop. Het hoge niveau op de uitgang van de eerste flipflop wordt ook doorgegeven aan een ingang van de NOF-poort, waardoor de uitgang van deze poort laag wordt. Bij **de stijgende flank van de tweede klokpuls** wordt het hoge niveau aan de ingang van de tweede flipflop doorgegeven naar de uitgang van de tweede flipflop. Tevens wordt het lage niveau aan de D-ingang van flipflop 1 doorgegeven naar zijn uitgang. **Bij elke volgende stijgende flank** van de klok schuift deze puls door naar de uitgang van de volgende flipflop. Zodra de uitgangen van de eerste drie flipflops allen nul zijn, wordt de uitgang van de NOF-poort weer hoog. Vanaf dan is er weer een puls beschikbaar op de ingang van de eerste flipflop die bij elke stijgende flank van de klok naar de volgende flipflop opschuift. Op deze manier blijft die puls oneindig lang door de uitgangen van de flipflops lopen.

Druk de schakeling en simulatie af in kleur (zie richtlijnen vroeger). Staan de nodenummers op de juiste plaats?

Sla de schakeling op als “Familienaam\_Voornaam\_Inleiding schakeling 5”.

Bij het opstarten van de schakeling gingen we ervan uit dat de uitgangen van de flipflops laag waren. Om zeker te zijn kunnen we een korte neergaande puls gebruiken (enkel op de PREB-ingang, niet op de CLRb-ingang) om de flipflops te resetten, maar het kan ook op een andere manier.

Klik op *Options → Global Settings...* en zet dan de parameter DIGINITSTATE op “1”. Alle latches en bistabiele elementen in de schakeling worden daardoor geset aan het begin van de simulatie (met een waarde “0” worden ze allemaal gereset, een “2” zorgt voor een willekeurige uitgangspositie). Druk op OK om te bevestigen.



Het resultaat van een 50 ms lange transiëntanalyse is te zien in Fig. 9. Alle uitgangen zijn in eerste instantie hoog, maar bij elke klokpuls wordt er eentje laag totdat er nog maar één over is. Dit hoge niveau blijft dan verder tussen de telleruitgangen circuleren.

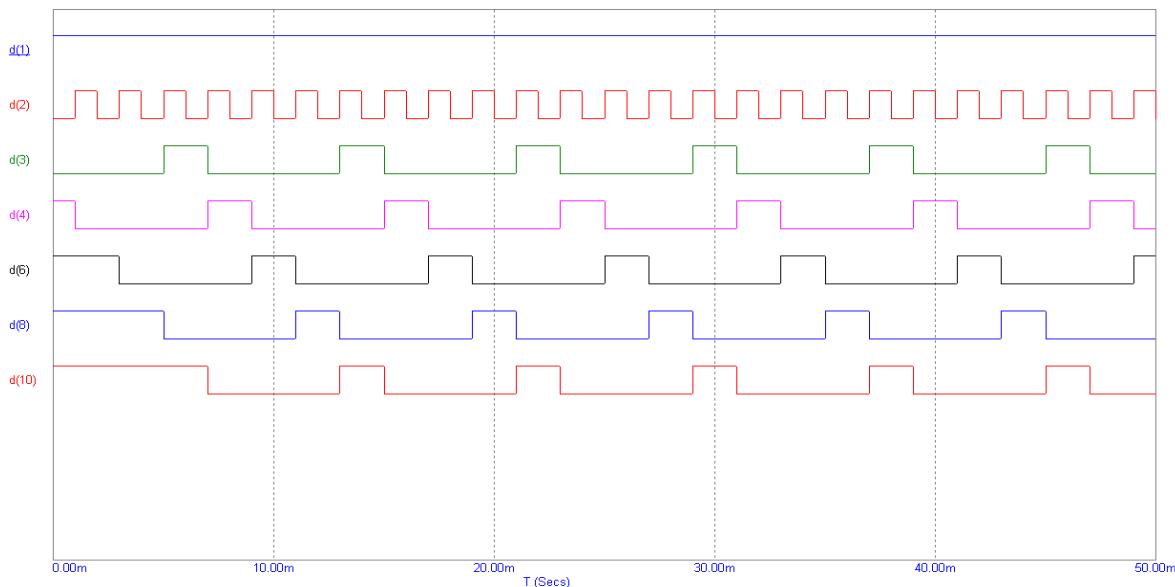


Fig. 9: De uitgangssignalen van de ringteller met DIGINITSTATE op 1

Vergelijk deze simulatie met de voorgaande analyse met DIGINITSTATE = 0.

Druk de schakeling en simulatie af in kleur (zie richtlijnen vroeger). Staan de nodenummers op de juiste plaats?

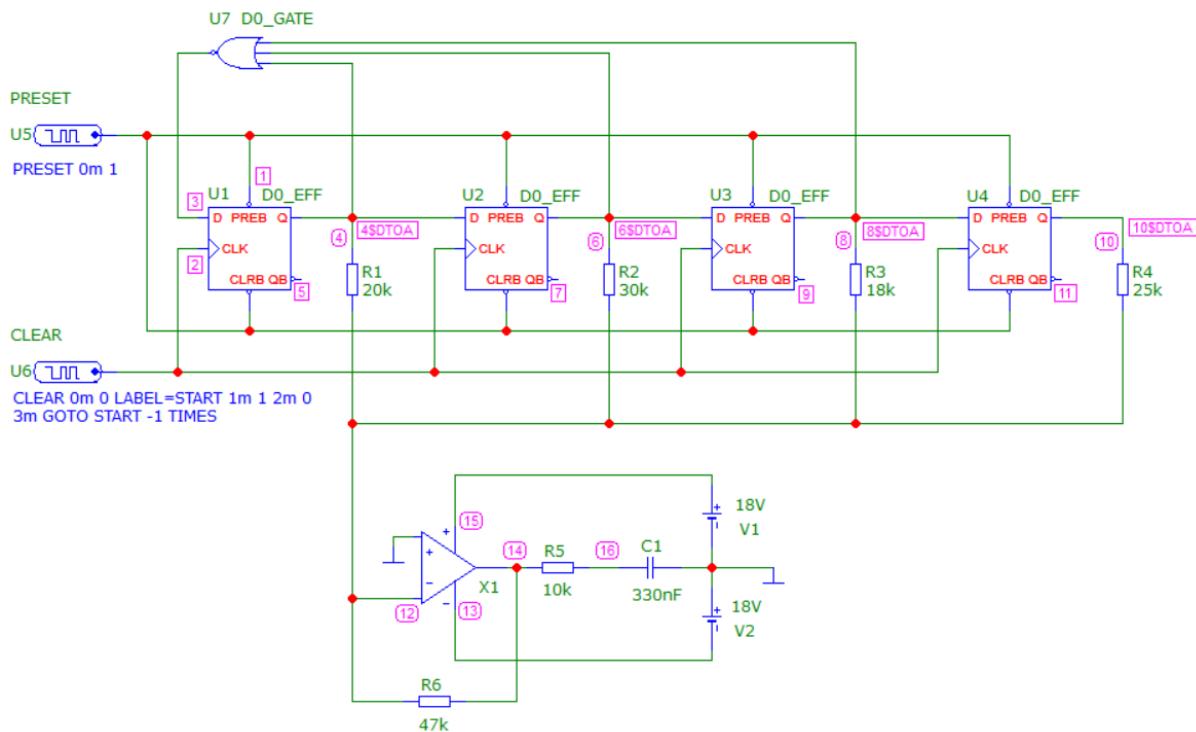
Sla de schakeling op als “Familienaam\_Voornaam\_Inleiding schakeling 6”.

## 1.11. Mixed mode

Sommige simulatieprogramma's zijn uitsluitend geschikt voor analoge schakelingen, terwijl andere alleen digitale elementen kunnen nabootsen. Micro-Cap en veel andere simulatieprogramma's van de laatste generatie zijn voor beiden geschikt; het is zelfs mogelijk **in een schema analoge en digitale componenten door elkaar te gebruiken**. Dit noemt men *mixed mode simulation*.

Als voorbeeld van zo'n situatie werken we verder op het voorgaande digitale circuit (ringteller) en voegen we een analoge schakeling toe die een analoge golfvorm produceert. Met meer flipflops zou een nauwkeurigere analoge golfvorm geproduceerd kunnen worden.

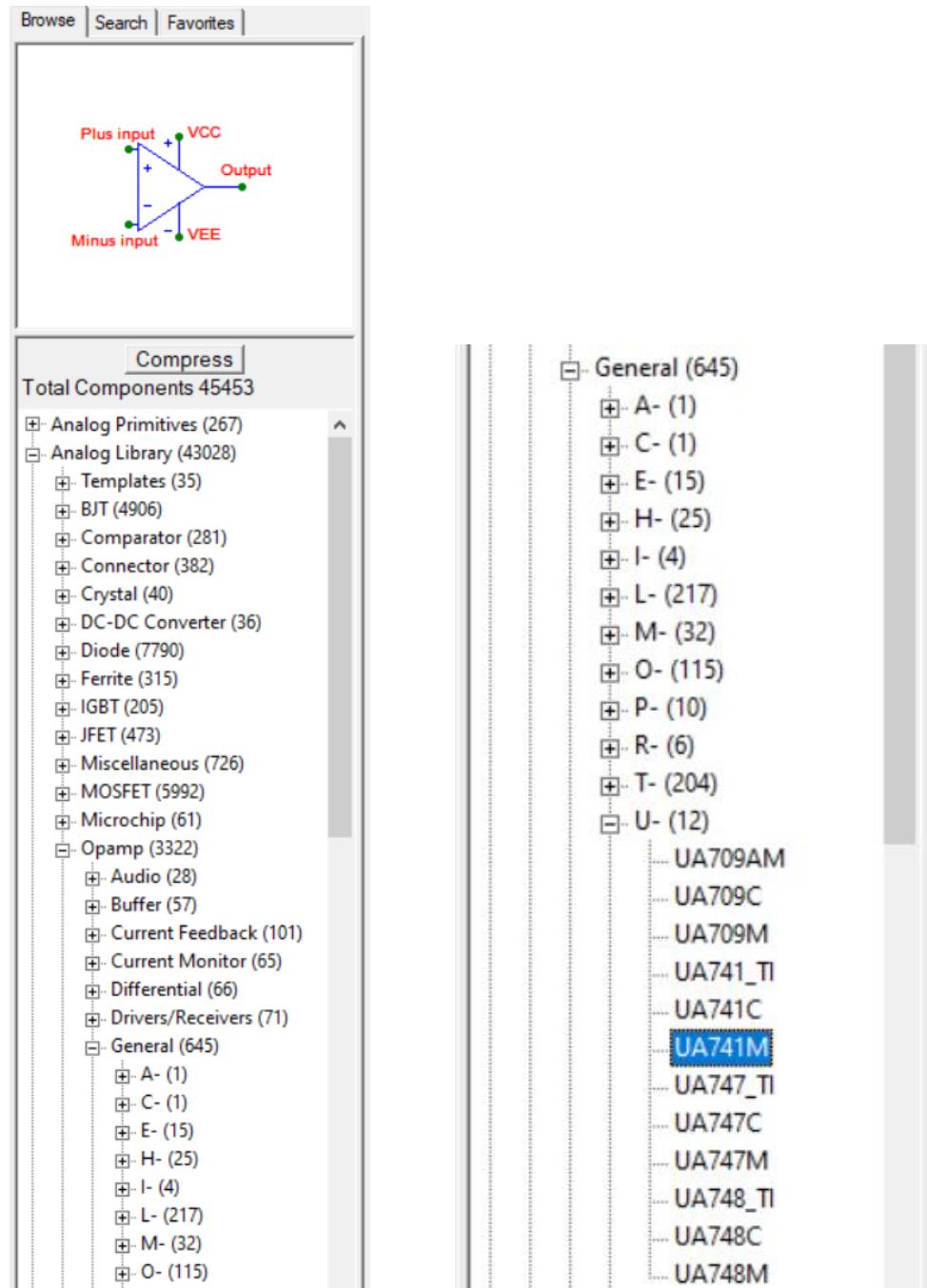
De te tekenen schakeling ziet er als volgt uit:



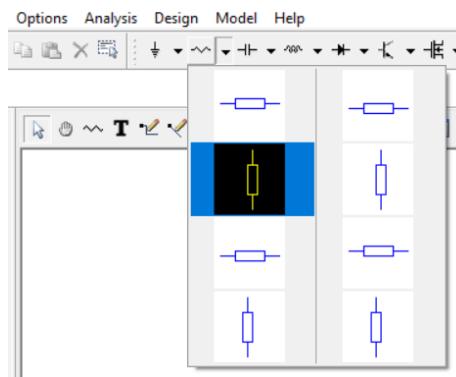
**Fig. 10:** Aan het schema uit Fig. 8 is een sommeerschakeling toegevoegd

We gaan eerst alle bijkomende componenten ophalen en op het schema plaatsen. Nadien leggen we de verbindingen.

De toe te voegen sommeerschakeling krijgt stromen van verschillende grootte toegevoerd vanuit de uitgangen van de teller. In Fig. 10 zijn alle uitgangen via weerstanden verbonden met de ingang van een UA741M-opamp. Kies de **opamp** via de menu's: Component/Analog Library/Opamp/General/U-/UA741M of via het panel links op het scherm:



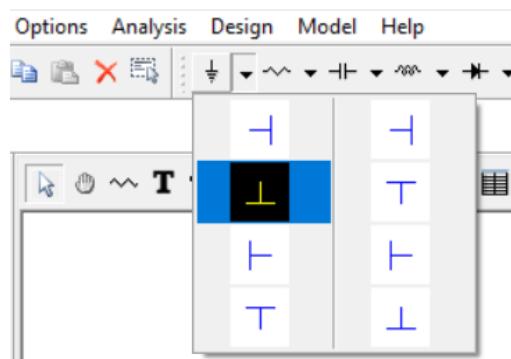
De weerstandswaarden bepalen de stroom die iedere uitgang levert. Je kan de **weerstanden** snel toevoegen via de pictogrammen bovenaan het scherm (druk op het pijltje rechts naast het weerstandssymbool). Kies de gewenste oriëntatie van de component: de eerste kolom zet R1 boven de waarde van de weerstand, de tweede kolom doet dit in omgekeerde volgorde. Achteraf kan je deze waarden nog steeds verplaatsen. Ook het symbool zelf kan geroteerd worden via de geziene weg. Geef telkens de juiste weerstandswaarde in **zonder een spatie** tussen de getalwaarde en de k van kilo! Ingave op de foutieve manier is mogelijk, maar zal een foutmelding opleveren bij het aanroepen van de transiëntanalyse (↔ bij EAGLE verlangen we wel een spatie tussen de getalwaarde en het voorvoegsel k).



Plaats op dezelfde wijze de **condensator**. Geef de waarde in **zonder een spatie** tussen de getalwaarde en de nF! Ingave op de foutieve manier is mogelijk, maar zal een foutmelding opleveren bij het aanroepen van de transiëntanalyse.

De **batterijen** V1 en V2 zijn ingesteld op 18V en leveren de voedingsspanning voor de opamp. Geef de waarde in **zonder een spatie** tussen de getalwaarde en de V! Ingave op de foutieve manier is mogelijk, maar zal een foutmelding opleveren bij het aanroepen van de transiëntanalyse. Let op de polariteit van de batterijen en controleer of hun ingestelde spanningen zichtbaar zijn (anders *Show* naast *Value* aanklikken)!

Plaats de **twee massasymbolen** via de pictogrammen bovenaan het scherm of via het menu door te klikken op Component/Analog Primitives/Connectors/Ground



Bij de knooppuntweergave in figuur 10 merken we dat:

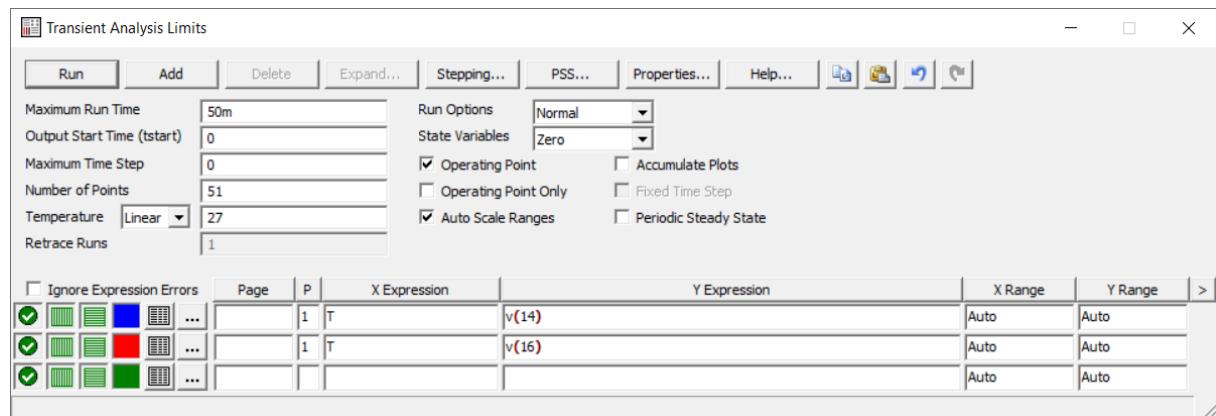
- in de verbindingen tussen het digitale en analoge deel automatisch digitaal naar analoog (= D/A) omzetters worden opgenomen. Dit wordt aangeduid door #DTOA waarbij # het getal van het knooppunt voorstelt. Dit zijn geen normale D/A-omzetters, maar **interfaces** tussen de secties die door Micro-Cap automatisch worden toegevoegd voor het verbeteren van de simulatieresultaten.
- digitale knooppunten rond het knooppuntnummer rechte hoeken heeft en afgeronde hoeken bij analoge knooppunten. Het niveau van digitale signalen kunnen we zoals reeds gezien voorstellen door d(#), de spanning van analoge signalen door v(#) waarbij # in beide gevallen het knooppunt (= node) voorstelt.

**Vooraleer we aan de simulatie beginnen, controleren we dat DIGINITSTATE = 0!**

In het scherm van de *Transient Analysis Limits* verwijderen we de lijnen die niet nodig zijn

**Delete**

door op de lijn te gaan staan en op **Delete** te drukken. Maak de curve van v(14) en v(16) zichtbaar. Dit is respectievelijk de uitgang van de opamp en de uitgang van de RC-filter. Het hier gesimuleerde circuit is een vereenvoudigde versie van een functiegenerator-IC. Door het aantal trappen te vergroten en een nauwkeurige dimensionering van de weerstanden kunnen allerlei golfvormen (zoals sinus en driehoek) heel natuurgetrouw worden nagebootst.

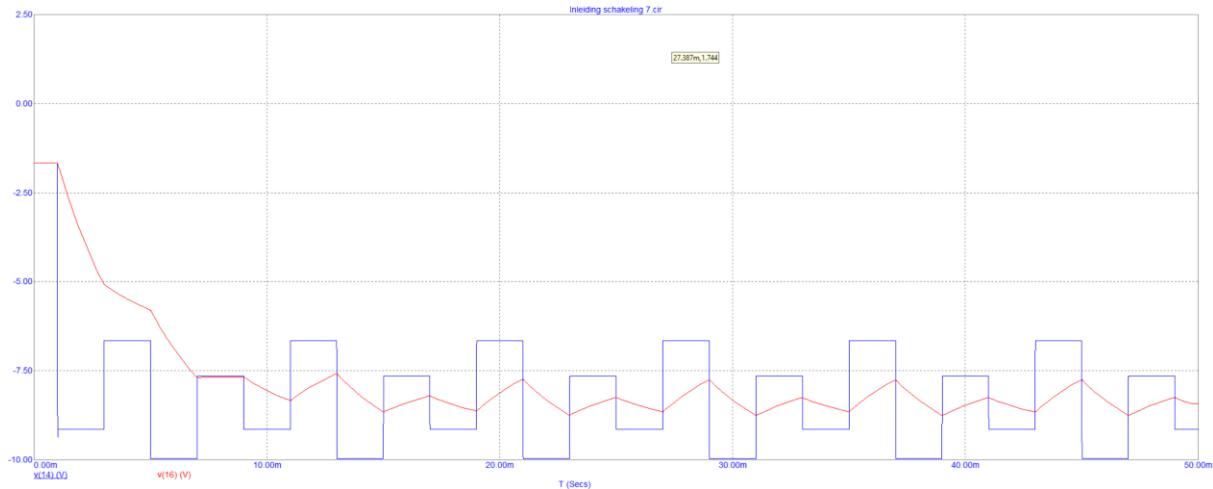


Je kan best op **X Range** en dan op Auto drukken om de X-as automatisch te laten aanpassen aan de berekende waarden.

Y Expression	X Range	Y Range
v(14)	Auto	-10,2.5
v(16)	Auto	-10,2.5
	TMAX,TSTART	to
	0.05,0,0.01	

Herhaal dit voor **Y Range**.

Fig. 11 toont de 50 ms lange transiëntanalyse van het signaal aan de uitgang van de opamp v(14). Bij elk kloksignaal verandert de grootte van het signaal (= spanning) omdat dan een andere weerstandswaarde met de opamp verbonden wordt. De aldus verkregen golfvorm (blauw) is weliswaar analoog van aard, maar ze vertoont nog steeds de karakteristieke steile flanken eigen aan een logische schakeling. Nadat het signaal door de RC-filter is gevoerd, lijkt het wat meer op een driehoek (rood).



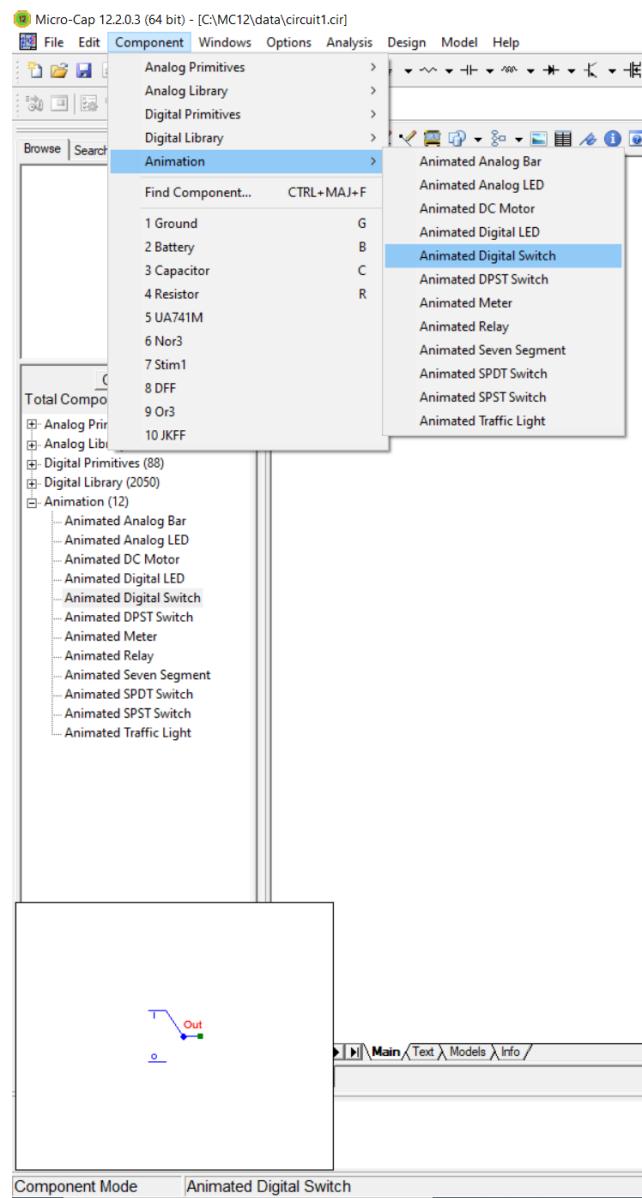
**Fig. 11: Het simulatieresultaat**

Kijk het schema nog eens na en verplaats namen, waarden en nodenummers indien nodig om een mooi geheel te verkrijgen (neem figuur 10 als leidraad). Druk de schakeling en simulatie af in kleur (zie richtlijnen vroeger).

Sla de schakeling op als “Familienaam\_Voornaam\_Inleiding schakeling 7”.

## 1.12. Het gebruik van animatie-objecten

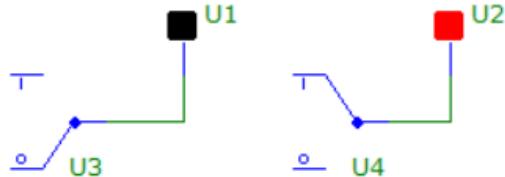
Normaal worden simulaties in Micro-Cap in één keer berekend. Er zijn echter een aantal hulpmiddelen voorhanden tijdens de simulatie om de ingangen te beïnvloeden en de toestand van de uitgangen te zien. Deze elementen kunnen teruggevonden worden onder de optie Component → **Animation** in de componentenbibliotheek of in het panel links van het scherm.



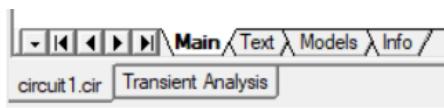
Een **Animated Digital Switch** is een schakelaar die met een klik van de muis verzet kan worden tussen 0 en 1 **enkel tijdens een simulatie**.



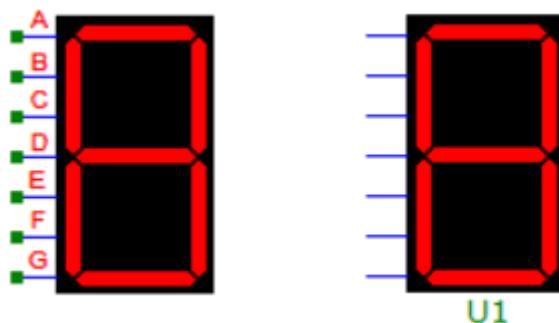
Een **Animated Digital LED** is een animatie-object dat zwart is als het aangesloten wordt op een logische 0 en rood wordt als het aangesloten wordt op een logische 1. In tegenstelling tot een werkelijke LED moet er geen voorschakelweerstand opgenomen worden tussen de digitale uitgang en de aangesloten LED, wordt er ook geen voeding getekend in het schema en is er slechts één “aansluitpootje”! Teken onderstaande schakeling:



Start de simulatie, ga naar het schema en dubbelklik op een schakelaar om deze te verzetten. Micro-Cap springt naar het simulatievenster zodat je onmiddellijk de invloed van je actie kan zien. Wil je teruggaan naar het schema, dan kan dat eenvoudig door onderaan het scherm op het tabblad *circuit1.cir* te klikken.



Een **Animated Seven Segment** bevat 7 aansluitingen en heeft dezelfde functie als een reëel 7-segment display. Dit display is echter van **het common-kathode type**, dit in tegenstelling tot het display dat in het labo gebruikt wordt, wat van het common-anode type is. Een dergelijk display kan je in Micro-Cap o.a. aansturen met een 7448-decoder (zie later). In het panel links op het scherm zie je de aansluitingen a ... g van het display. Een led licht op als zijn overeenkomstige aansluiting een hoog niveau heeft.

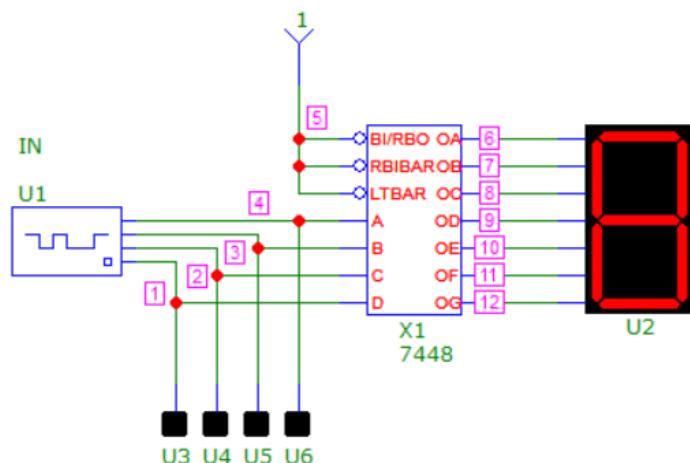


Aansluitingen in het panel (links) en schemasymbool (rechts)

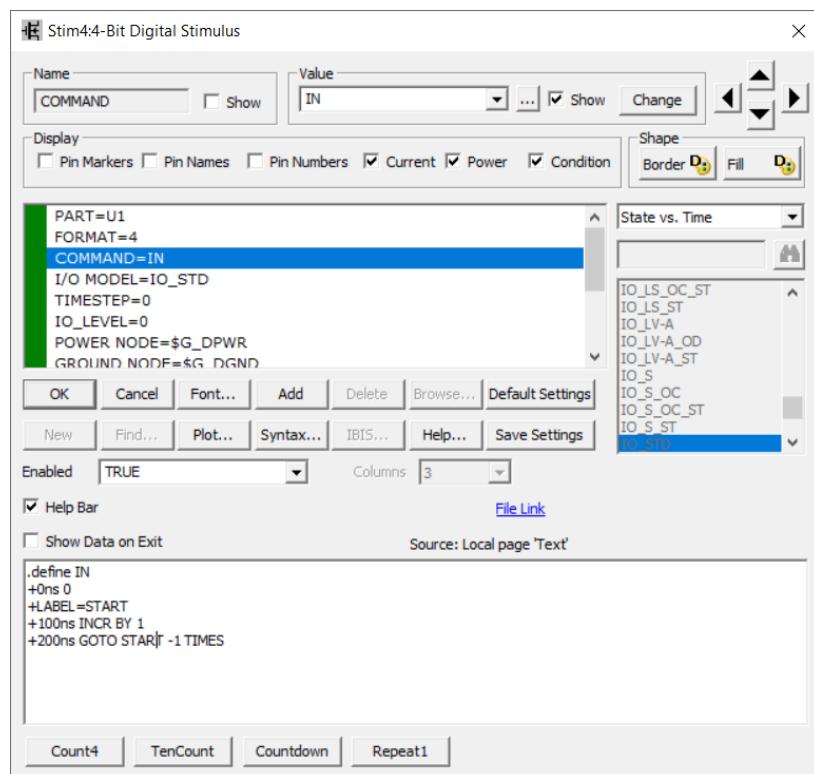
### 1.13. Stap per stap simuleren

Wanneer we bv. een 7-segment display willen simuleren, dan willen we er ons van vergewissen dat alle ingangscodes foutloos getoond worden op ons display. In dat geval willen we de ingangscode kunnen instellen en zien wat het display vertelt.

Eenmaal je de *Animation*-componenten opgenomen hebt in je schema kan je een simulatie uitvoeren zonder stoppen (zie vorige paragrafen). Om echter de simulatie stap per stap uit te kunnen voeren moet je de volgende stappen ondernemen. Bekijk hierbij aandachtig het schema hieronder.



De bedoeling is om de werking van een 7-segmentdecoder (7448) na te gaan. Wij gebruiken daarvoor een **4-bit stimulusgenerator** met de volgende instellingen:



Geef aan *Value* de naam IN.

Stel **FORMAT = 4** (in plaats van 1111).

Merk op dat hier voor de instelling van COMMAND verwezen wordt naar het programma ‘IN’. Vervolledig onderaan de inhoud van dit programma, er staat al: *.define IN*. Merk ook de andere manier van noteren met +.

```
.define IN
+0ns 0
+LABEL=START
+100ns INCR BY 1
+200ns goto start -1 TIMES
```

We starten met 0 op 0ns, dan komt het LABEL=START en dan verhogen we na 100ns de waarde met 1 (incrementeer met 1). De lus wordt gesloten op 200ns en wordt oneindig keer doorlopen (- 1 TIMES).

De uitgangen van de 4-bit stimulusgenerator houden we in de gaten met **4 digitale LED’s**.

Plaats het **decoder-IC 7448** via het menu: Component/Digital Library/74xx42-/42-/7448. Ga naar het pijltje (*Select Mode*) en dubbelklik op de component zodat het scherm met de instellingen opent. Klik bij *Value* de box *Show* aan zodat het type IC ook in het schema wordt weergegeven.

De uitgangen van de generator verbinden we dan sequentieel met de A, B, C en D-ingangen van de 7448. De uitgangen van de 7448 komen perfect overeen met de aansluitingen van het 7-segment LED-display zodat we deze rechtstreeks kunnen verbinden.

Er zijn nog 3 ingangen:

- *BI/RBO* wat staat voor *Blanking Input / Ripple Blanking Output*
- *RBIBAR* wat staat voor *Ripple Blanking Input*
- *LTBAR* wat staat voor *Lamp Test*

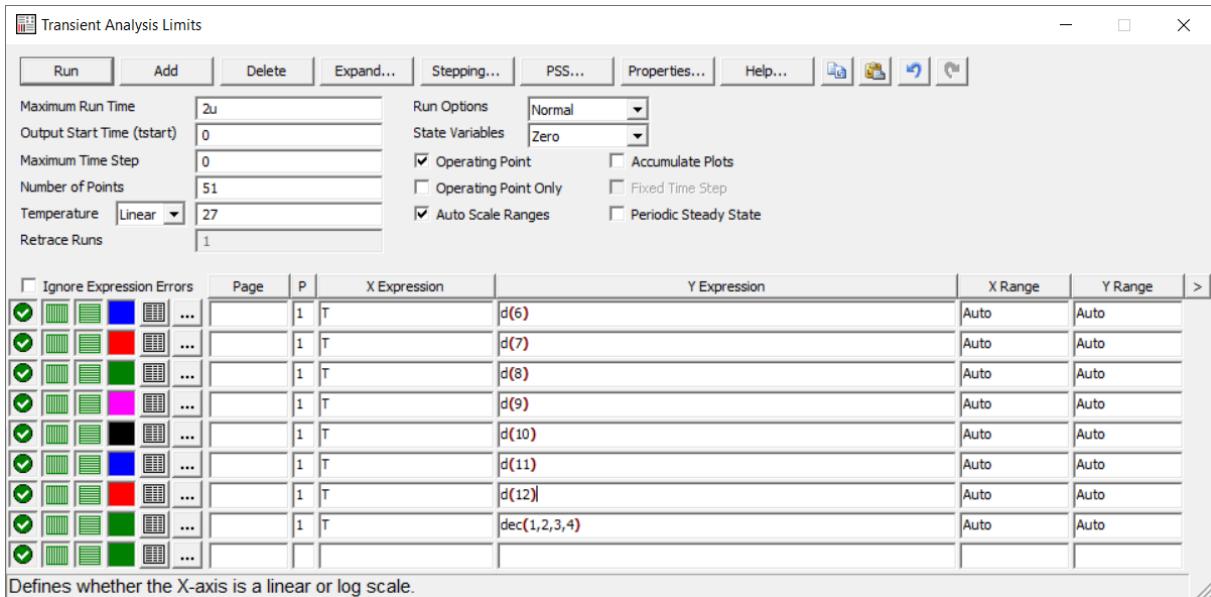
De in-/uitgang BI/RBO leggen we aan een hoog niveau. Dit gebeurt met een digitale stimulus generator **Fixed Digital** die op 1 staat. Je haalt deze component op via het menu: Component/Digital Primitives/Stimulus Generators/Fixed Digital en geef deze *Value* = 1. BI en RBI worden gebruikt om bij serieschakeling van zevensegmentaanduidingen de meer beduidende cijfers te *blanken* (=doven) als ze op 0 decimaal staan (zie theorie later).

Vermits we RBIBAR en LTBAR niet gebruiken, verbinden we deze op dezelfde wijze met 1 om ze zeker inactief te maken (je mag geen zwevende ingangen hebben).

Verbind alle geplaatste componenten volgens het schema.

Sla de schakeling op als “Familienaam\_Voornaam\_Inleiding schakeling 8”. Micro-Cap durft immers bij de volgende simulatie wel eens vast te lopen...

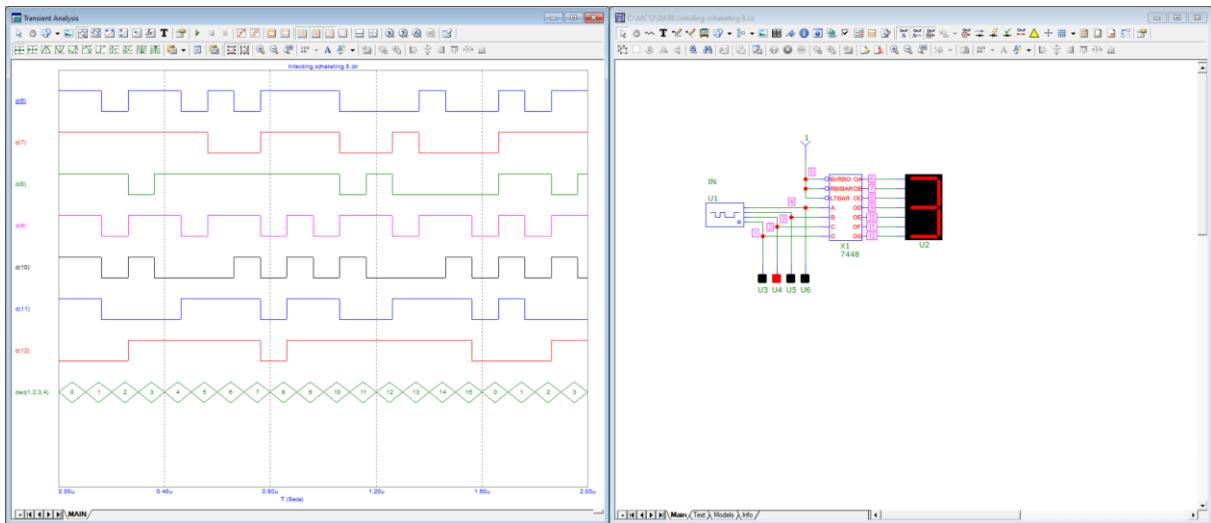
## Instelling transiënt analyse



*Maximum Run Time* staat op  $2 \mu\text{s}$  om alle mogelijke situaties zichtbaar te maken en een stukje dat zich herhaalt.

Wij gaan nu de uitgangen van de 7-segmentsdecoder d(6) tot d(12) zichtbaar maken, de ingangen gaan we voor de duidelijkheid verzamelen tot een decimale waarde via  $\text{dec}(1,2,3,4)$ .

Om een overzicht te hebben, plaats je **de 2 vensters van schema en analyse naast elkaar** via **Windows/Tile Vertical**. Als we Run indrukken krijgen we onmiddellijk het volledige analysevenster en zien we de waarde 3 op het 7-segment LED display en de ledaanduidingen van de ingang. Deze waarde stemt overeen met de waarde op het einde van de simulatie (dit hadden we daarstraks ingesteld op  $2 \mu\text{s}$ ). Controleer deze waarde in het simulatievenster!

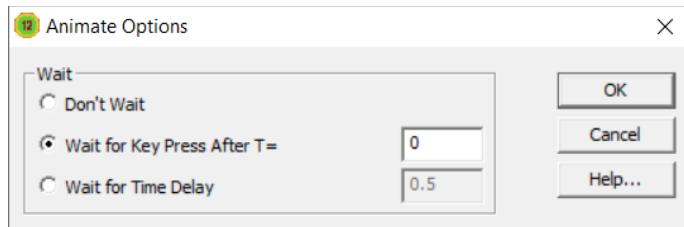


Bemerk dat de pulscombinatie bij de start niet juist is. Daarentegen zijn de uitgangen van de decoder wel laag als de decimale waarde voor de tweede keer de waarde nul aanneemt.

Je kan ook de **analyse stap voor stap** uitvoeren. Zodra de transiëntanalyse gestart is, wordt de menuoptie **Scope/Animate options...** zichtbaar (of via 

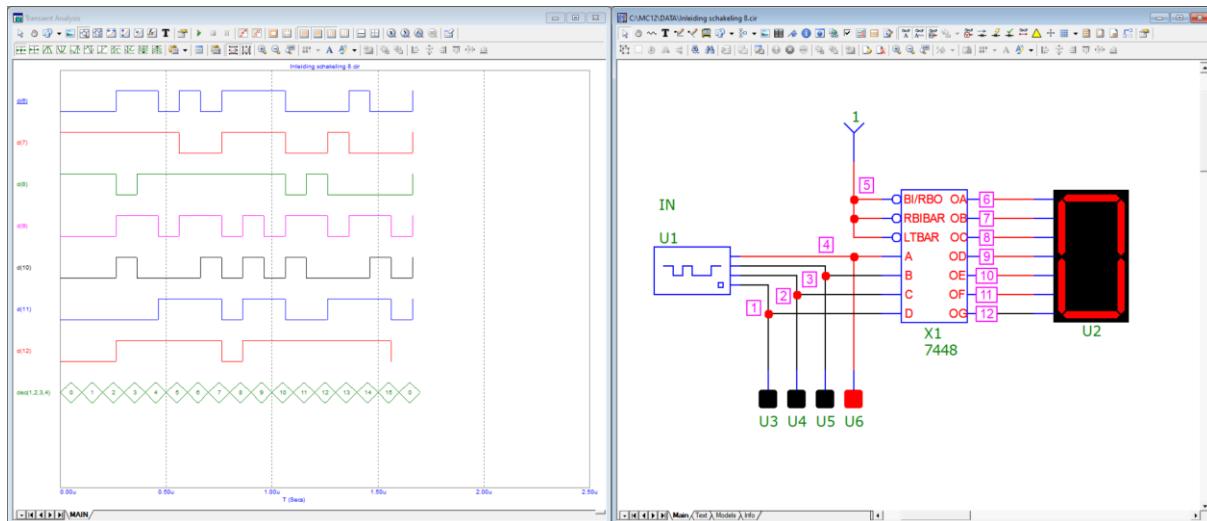


Kies de optie *Wait for Key Press After T=*. Druk op OK.



Start de simulatie opnieuw (F9).

Door nu verschillende keren op **CTRL+spatiebalk** te drukken kan je stap voor stap iedere code bekijken. Je zal zien dat pulsdiagramma's pas afgebeeld worden als er een overgang plaats vindt. In het schema geeft het 7-segmentdisplay de juiste decimale waarde weer.

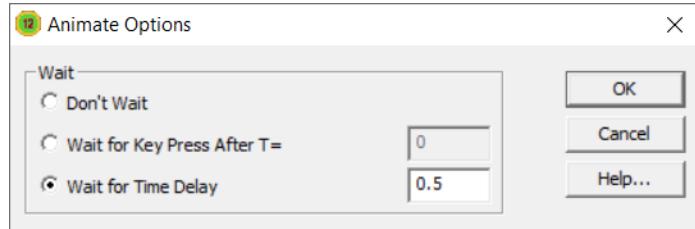


Indien nodig, kan je de simulatie op elk gewenst moment onderbreken door op de ESC-toets te drukken. Of je kan de simulatie tijdelijk of definitief onderbreken met de “**PAUSE**” of de “**STOP**”-knop.



Kijk het schema nog eens na en verplaats namen, waarden en nodenummers indien nodig om een mooi geheel te verkrijgen. **Stop de simulatie bij de waarde die je verjaardagsmaand voorstelt.** Druk de schakeling met deze simulatie af in kleur (zie richtlijnen vroeger).

Je kan in **Scope/Animate Options ...** ook kiezen voor de optie **Wait for Time Delay**. Je simulatie wordt nu automatisch uitgevoerd maar er wordt telkens een tijdje gewacht op de plaats waar je in het eerste geval nog op een toets moest drukken.



Druk op of F2 (Run) om de simulatie te starten.

Als variant kan je ook digitale schakelaars voorzien in plaats van de digitale stimulus Fixed Digital. Op die manier kan je door deze om te schakelen (dubbelklikken op het schakelend deeltje van de switch) het effect bekijken van RBIBAR of LTBAR (maar enkel tijdens de analyse zelf). Probeer dit zelf uit op vrije basis.

## 1.14. Interessante weetjes

Bij het tekenen van grafieken kunnen een aantal interessante functies gebruikt worden bij digitale grootheden. Deze functies kunnen gebruikt worden in het veld “Y Expression” van het “*Transient Limits Analysis*” venster wanneer we de gegevens voor een simulatie invullen met b3 het most significant bit (MSB) en b0 het least significant bit(LSB):

DEC (b3, b2, b1, b0)	Toont de decimale waarde ( $0 \rightarrow 9$ en ook verder van $10 \rightarrow 15$ ) van het digitaal getal gevormd door de uitgangen b3 t.e.m. b0
BIN (b3, b2, b1, b0)	Toont de binaire waarde van het digitaal getal gevormd door de uitgangen b3 t.e.m. b0
HEX (b3, b2, b1, b0)	Toont de hexadecimale waarde ( $0 \rightarrow F$ ) van het digitaal getal gevormd door de uitgangen b3 t.e.m. b0
+ of -	Neemt de som of het verschil van twee decimale, binaire of hexadecimale getallen (gedefinieerd als hierboven)
&,  , ^, ~	Logische functie op bitniveau van twee binaire signalen (EN, OF, XOF, NIET)