

# PHASE TO PHASE

## Mogelijkheden met macro's

04-141 pmo

22 december 2004

Phase to Phase BV  
Utrechtseweg 310  
Postbus 100  
6800 AC Arnhem  
T: 026 356 38 00  
F: 026 356 36 36  
[www.phasetophase.nl](http://www.phasetophase.nl)

© Phase to Phase BV, Arnhem, Nederland. Alle rechten voorbehouden.

Dit document bevat vertrouwelijke informatie. Overdracht van de informatie aan derden zonder schriftelijke toestemming van of namens Phase to Phase BV is verboden. Hetzelfde geldt voor het kopiëren van het document of een gedeelte daarvan.

Phase to Phase BV is niet aansprakelijk voor enige directe, indirecte, bijkomstige of gevolgschade ontstaan door of bij het gebruik van de informatie of gegevens uit dit document, of door de onmogelijkheid die informatie of gegevens te gebruiken.

## INHOUD

1	Inleiding .....	4
2	Nieuwe comando's .....	4
3	Nieuwe rekenkundige functies .....	5
4	Nieuwe tekstfuncties .....	6
5	Macro-editor .....	7
6	Nieuwe attributen .....	8
7	Voorbeeld 1: Omschakelen in een distributienet .....	8
8	Voorbeeld 2: willekeurig gedrag wind .....	10
9	Voorbeeld 3: uitwisseling vermogen .....	13
10	Voorbeeld 4: dwarsregelaar .....	14
11	Conclusie.....	16

## 1 INLEIDING

De macro's in Vision bieden de mogelijkheid om automatisch repeterende bewerkingen op een net-model uit te voeren. Zo kunnen diverse tijdrovende studies via de krachtige macrotaal geautomatiseerd worden. De resultaten kunnen als tekst of in een spreadsheet gerapporteerd worden. Aan de hand van een viertal voorbeelden wordt het gebruik van de macro's toegelicht.

De inhoud van dit rapport is gepresenteerd op de Vision gebruikersdag op 14 december 2004.

## 2 NIEUWE COMANDO'S

- **Fault**
  - Uitvoeren sequentiële storingsanalyse
- **Copy**
  - Kopiëren reeks cellen in spreadsheet
- **GetTitle**
  - Opvragen naam spreadsheet blad

**Fault:** Uitvoeren van de sequentiële storingsanalyseberekening naar keuze op één knooppunt of op alle knooppunten.

Indien de berekening voor één knooppunt wordt aangeroepen, wordt voor dat knooppunt de kortsluiting berekend en zijn van de andere knooppunten en takken de spanningen en stromen in die situatie beschikbaar. De transformatortrapstanden worden bij deze berekening niet aangepast aan de loadflowsituatie.

Indien de berekening voor alle knooppunten wordt aangeroepen, wordt alleen de initiële situatie (door de loadflow bepaald) uitgerekend. De foutsoort en de foutimpedantie moeten dan wel worden gespecificeerd, maar hebben dan geen invloed op de berekening.

Algemene notatie:

Fault ( **Node**( *ObjectNaam* ), Sluiting, Rf, Xf )

of

Fault ( NIL, Sluiting, Rf, Xf )

Op de plaats van Sluiting moet een tekstwaarde worden ingevuld of een tekstvariabele. De mogelijkheden zijn:

'abc', 'abco', 'ao', 'bo', 'co', 'ab', 'ac', 'bc', 'abo', 'aco', 'bco'.

Rf: de waarde van de weerstand op de foutplaats, in Ohm.

Xf: de waarde van de reactantie op de foutplaats, in Ohm

**Copy:** kopiëren van een reeks cellen van een blad van een spreadsheet naar een ander blad van de output-spreadsheet.

Algemene notatie:

*Copy( FileIndex, bladA , rij1 , kolom1 , rij2 , kolom 2 , bladB , rij3 , kolom 3 )*

FileIndex is de index die verwijst naar het met behulp van Open geopende spreadsheetbestand. Indien FileIndex de waarde 0 toegewezen krijgt, kan worden gekopieerd uit het spreadsheetbestand dat door de Macro met Write wordt beschreven.

Het resultaat komt altijd terecht in de output-spreadsheet (met FileIndex nummer 0).

**Gettitle:** Opvragen van een naam van een blad in het werkboek onder FileIndex. Het blad is alleen bereikbaar via het bladnummer.

Algemene notatie:

*GetTitle ( FileIndex, bladnummer, titel )*

FileIndex is de index die verwijst naar het met behulp van Open geopende spreadsheetbestand. Indien FileIndex de waarde 0 toegewezen krijgt, wordt verwezen naar het spreadsheetbestand dat door de Macro met Write wordt beschreven.

### 3 NIEUWE REKENKUNDIGE FUNCTIES

- Max
  - het maximum van twee waarden
- Min
  - het minimum van twee waarden
- Mid
  - het gemiddelde van twee waarden
- Dif
  - het absolute verschil van twee waarden
- Power
  - machtsverheffen
- Random
  - willekeurig getal

MAX: het maximum van twee waarden;      bijvoorbeeld Text(MAX(1,2) levert: 2

MIN: het minimum van twee waarden;      bijvoorbeeld Text(MIN(1,-2) levert: -2

MID: het gemiddelde van twee waarden;      bijvoorbeeld Text(MID(1,2) levert: 1.5

DIF: het absolute verschil van twee waarden;      bijvoorbeeld Text(DIF(1,-2) levert: 3

POWER: machtsverheffen;                    bijvoorbeeld `Text(Power(2,5))` levert: 32

RANDOM: genereert een willekeurig getal tussen 0 en Max (geheel getal);

bijvoorbeeld

`Set(Max,1000000)`

`Text(RANDOM(Max))`

levert een getal als: 765836

#### 4                    NIEUWE TEKSTFUNCTIES

- Uppercase, Lowercase
  - Conversie naar hoofd- en kleine letters
- Trim
  - Verwijderen begin- en eindspaties
- Leftstr, Rightstr, Midstr
  - Tekens van een tekststring
- Length
  - Lengte van een tekststring
- Pos
  - Positie van een tekst in een tekststring

UPPERCASE: Conversie naar hoofdletters;    `Text(UPPERCASE('abc'))` levert: ABC

LOWERCASE: Conversie naar kleine letters;    `Text(UPPERCASE('ABC'))` levert: abc

TRIM: Verwijderen van begin- en eindspaties;    `Text(TRIM(' abc '))` levert: abc

LEFTSTR: Linkertekens van een string;    `Text(LEFTSTR('abc',2))` levert: ab

RIGHTSTR: Rechtertekens van een string;    `Text(RIGHTSTR('abc',2))` levert: bc

MIDSTR: Tekens van het midden van een string;    `Text(MIDSTR('abcde',2,3))` levert: bcd

LENGTH: Aantal tekens van een string;    `Text(LENGTH('abc'))` levert: 3

POS: Positie van een tekst in een string;    `Text(POS('bc','abcde'))` levert: 2

5 MACRO-EDITOR

- Code inside
- Code completion
- Indenten met Shift-Ctrl-I
- Unindenten met Shift-Ctrl-U

The screenshot shows a window titled 'Macro Uitwisseling.vmf' with a menu bar (Bestand, Bewerken, Macro, Help) and a toolbar. The code editor contains the following text:

```

10  set (doorgaan, true)
11  while (doorgaan, =, true)
12      if ( source('EC').P , > , Puit )
13      add (
14      load Variabele, Waarde
15      if ( Object(ObjectNaam).Attribuut, Waarde)
16          set (doorgaan, false)
17      end
18      if ( syngen('Site C Local Gen').Pref , >

```

A yellow tooltip box is positioned over the text 'Object(ObjectNaam).Attribuut, Waarde' on line 15.

Code inside

The screenshot shows the same window as above, but with a different code structure and a dropdown menu open. The code editor contains:

```

10  set (doorgaan, true)
11  while (doorgaan, =, true)
12      if ( source('EC').P , > , Puit )
13      add (syngen('Site C Local Gen').|
14      loadflow( 0, L, true )
15      if ( source('EC').P , < , Puit
16          set (doorgaan, false)
17      end
18      if ( syngen('Site C Local Gen')
19          set (doorgaan, false)
20      end
21      else

```

A dropdown menu is open over the code, listing the following items:

- Name: Text
- Sort: Text
- Specifics: Text
- Selected: Boolean
- Group: Integer
- Node: Text
- Sheet: Text
- X: Integer

Code completion

## 6 NIEUWE ATTRIBUTEN

- Beeldgegevens van alle objecten
  - Color, thickness, style, size
- Coördinaten van de takken
  - XY
- Hulpattributen bij alle objecten
  - Bool, real, string, object

## 7 VOORBEELD 1: OMSCHAKELEN IN EEN DISTRIBUTIENET

- Doel:
  - Vinden van de hoogst belaste kabel bij het omschakelen in een distributienet met open ringen
- Werkwijze:
  - Alle netopeningen sluiten
  - Alle kabels een keer laten uitvallen
  - Maximale kabelbelasting bekijken

**Mogelijkheden met macro's, voorbeeld 1: Omschakeling**

```

write(1,1,1,'Kabelnaam')           //schrijf header
write(1,1,2,'Normale belasting')
write(1,1,3,'Vermaasde belasting')
write(1,1,4,'Maximale belasting bij uitval')
write(1,1,5,'In kabel')
fit(1,1,1,1,4)                     //fit kolombreedte
set(rij,1)
loadflow(0,,FALSE)
for(cable,eenkabel)                //doorloop ale kabels
  if(eenkabel.result,=,'LF')
    add(rij,1)
    write(1,rij,1,eenkabel.name)
    write(1,rij,2,eenkabel.load)    //schrijf normale belasting in 2e kolom

```



```

        set(eenkabel.real,rij)           //onthoud het rijnummer bij de kabel
    end
end
for(cable,eenkabel)
    set(eenkabel.firstswitch,closed)
    set(eenkabel.secswitch,closed)     //zet de schakelaars van alle kabels
    dicht
end
loadflow(0,,FALSE)
for(cable,eenkabel)
    if(eenkabel.real,>,0)
        if(eenkabel.result,=,'LF')
            write(1,eenkabel.real,3,eenkabel.load) //schrijf de 'vermaasde'
belasting
        end
    end
end
for(cable,uitvalkabel)
    if(uitvalkabel.real,>,0)
        set(uitvalkabel.firstswitch,open)
        set(uitvalkabel.secswitch,open)     //laat een kabel uitvallen door
//beide schakelaars te openen
        loadflow(0,,FALSE);                //doe een loadflow
        set(MaximaleBelasting,0)
        set(MaximaleKabel,NIL)
        for(cable,eenkabel)                 //doorloop alle kabels
            if(eenkabel.result,=,'LF')
                if(eenkabel.load,>,MaximaleBelasting) //als belasting hoger dan
//we al gehad hebben
                    set(MaximaleBelasting,eenkabel.load) //bewaars hoogste belasting
                    set(MaximaleKabel,eenkabel) //en de bijbehorende kabel
                end
            end
        end
        if(MaximaleBelasting,>,0)
            write(1,uitvalkabel.real,4,MaximaleBelasting) //rapporteer maximaal
//gevonden belasting in 4e kolom
            write(1,uitvalkabel.real,5,MaximaleKabel.name) //+ bijbehorende kabel
// in de vijfde kolom
        end
        set(uitvalkabel.firstswitch,closed)
        set(uitvalkabel.secswitch,closed) //sluit de schakelaars weer
    end
end
end

```

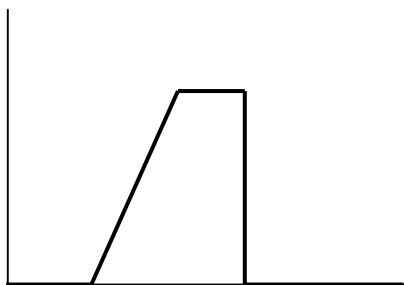
## 8 VOORBEELD 2: WILLEKEURIG GEDRAG WIND

- Doel:
  - Bekijken gevolgen voor de spanningskwaliteit
- Toepassing:
  - Normaal distributienet
  - Inpassing windpark

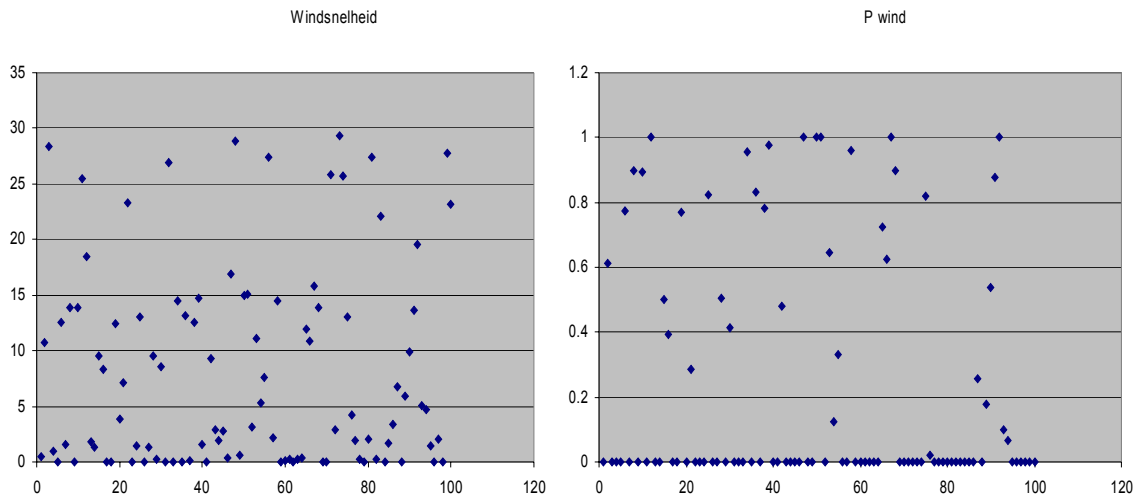
## Voorbeeld 2: gedrag windturbine

- Weibull-verdeling  $F(t) = 1 - e^{-\left(\frac{t}{\theta}\right)^b}$
- Eenvoudig model windturbine:

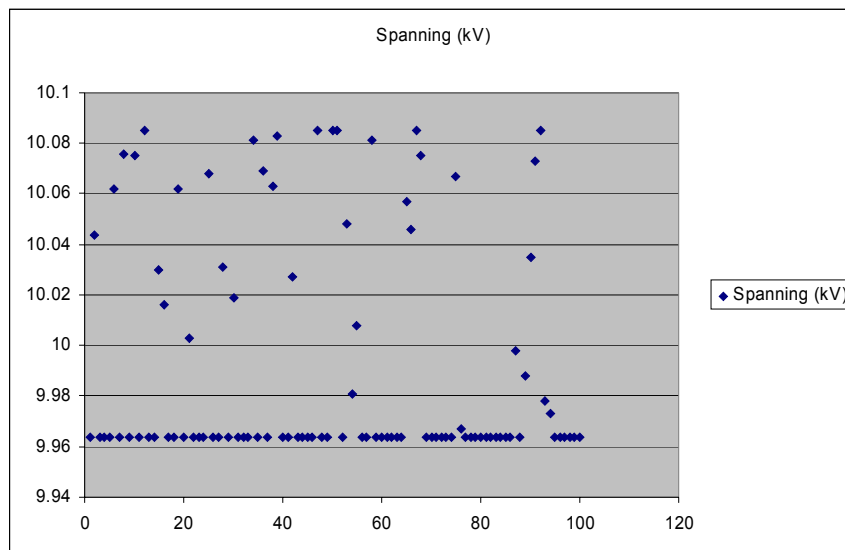
$$P_V(v) = \begin{cases} 0 & v \leq V_{ci} \\ P_r \cdot \frac{v - V_{ci}}{V_r - V_{ci}} & V_{ci} \leq v \leq V_r \\ P_r & V_r \leq v \leq V_{co} \\ 0 & v > V_{co} \end{cases}$$



## Voorbeeld 2: resultaat output wind



## Voorbeeld 2: resultaat spanning



### Mogelijkheden met macro's, voorbeeld 2: Inpassing windenergie

```
// Parameters door de gebruiker in te vullen //
set(vci,4) // windsnelheid inschakelen windturbine: 4 m/s //
set(vr,15) // nominale windsnelheid windturbine : 15 m/s //
set(vco,20) // maximum windsnelheid windturbine : 20 m/s //
set(Pnom,1) // windturbine Pnom : 1 MW //
////////////////////////////////////

set(theta,1) // voor de Weibull verdeling
set(b,4) // voor de Weibull verdeling
```

```

set(Max,1000000) // voor de random functie
// Berekening vermenigvuldigingsfactor
set(Factor,Pnom)
set(noemer,vr)
subtract(noemer,vci)
divide(Factor,noemer) // Factor := Pnom / (vr - vci)
// Header van de tabel
write(1,1,1,'n')
write(1,1,2,'Windsnelheid')
write(1,1,3,'P wind')
write(1,1,4,'Spanning (kV)')

// Doe 100 berekeningen
Loop ( teller, 1, 100, 1)
  set(t,RANDOM(Max))
  divide(t,Max)
  set(exponent,t)
  divide(exponent,theta)
  set(exponent,power(exponent,b))
  set(exponent,neg(exponent))
  set(emacht,exp(exponent))
  set(Weibull,1)
  subtract(Weibull,emacht)
  multiply(Weibull,50) // Windsnelheid volgens een Weibull kansverdeling
// Windmolenkarakteristiek
if(Weibull,<,vci)
  set(Pwind,0) // 0
else
  if(Weibull,<,vr)
    set(Pwind,Weibull) //
    subtract(Pwind,vci) // (v-Vci) x Factor
    multiply(Pwind,Factor) //
  else
    if(Weibull,<,vco)
      set(Pwind,Pnom) // Pnom
    else
      set(Pwind,0) // 0
    end
  end
end
end
// Afdrukken windturbine output
set(rij,teller)
add(rij,1)
write(1,rij,1,teller)
write(1,rij,2,Weibull:7:3)
write(1,rij,3,PWind:7:3)
// Instellen asynchrone machine en loadflowberekening
set(asyngen('Wind').Pmref,PWind)
loadflow(0,L,true)
write(1,rij,4,Node('MV node B4').U)
end
fit(1,1,1,Rij,4)

```

9 **VOORBEELD 3: UITWISSELING VERMOGEN**

- Doel:
  - Uitgewisseld vermogen gelijk aan gecontracteerd vermogen maken
- Werkwijze:
  - Industrienet met voorgeschreven vermogen op aankoppelingspunt
  - Eén lokale generator regelt actief vermogen
- Mogelijke uitbreidingen:
  - Gecombineerd P en Q uitwisseling
  - Regelvermogen verdelen over meerdere generatoren

**Mogelijkheden met macro's, voorbeeld 3: Uitgewisseld vermogen**

```

set(Puit,4)
set(Pmax,2)
set(Pmin,0)
loadflow( 0, L, true )
if(network.result,='LF')
  text('Uitgewisseld vermogen energiebedrijf: ',source('EC').P,' MW')
  text('Lokale opwekking: ',syngen('Site C Local Gen').P,' MW')
  text('Gewenste uitwisseling: ',Puit,' MW')

  set(doorgaan,true)
  while(doorgaan,=,true)
    if ( source('EC').P , > , Puit )          // uitgewisseld vermogen te groot
      add(syngen('Site C Local Gen').Pref,0.1)
      loadflow( 0, L, true )
      if ( source('EC').P , < , Puit )        // stoppen als P in orde
        set(doorgaan,false)
      end
      if ( syngen('Site C Local Gen').Pref , >= , Pmax )
        set(doorgaan,false)                 // stoppen als generator uitgeregeld
      end
    else                                       // uitgewisseld vermogen te klein
      subtract(syngen('Site C Local Gen').Pref,0.1)
      loadflow( 0, L, true )
      if ( source('EC').P , > , Puit )
        set(doorgaan,false)                 // stoppen als P in orde
      end
      if ( syngen('Site C Local Gen').Pref , <= , Pmax )
        set(doorgaan,false)                 // stoppen als generator uitgeregeld
      end
    end
  end
end

text('=====')
text('Nieuwe instelling:')
text('Uitgewisseld vermogen energiebedrijf: ',source('EC').P,' MW')
text('Lokale opwekking: ',syngen('Site C Local Gen').P,' MW')

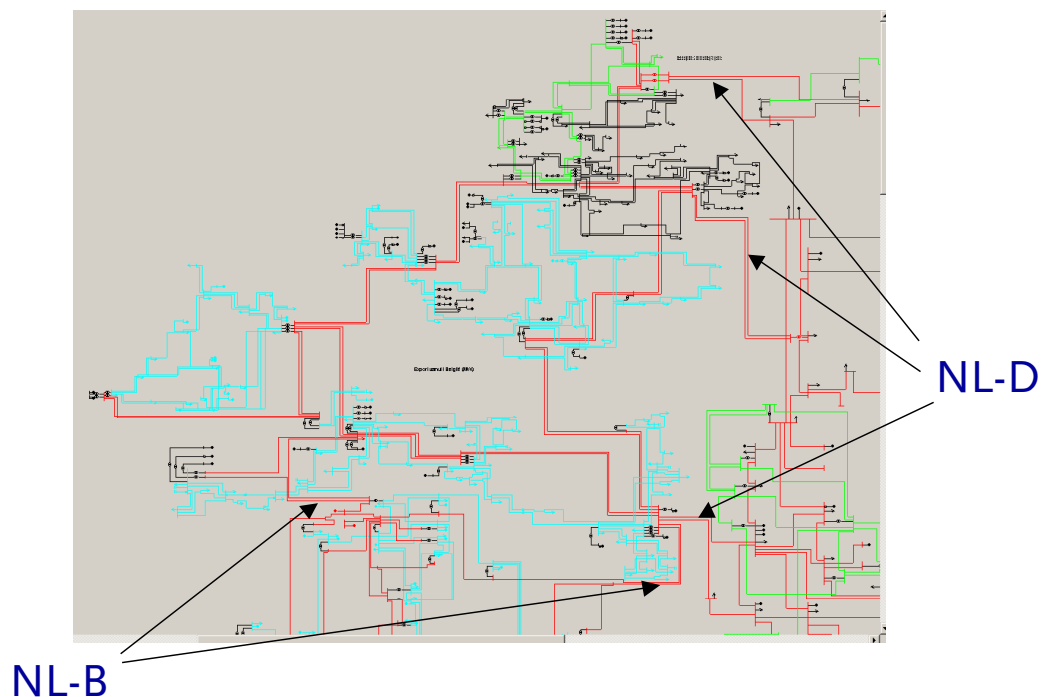
end

```

## 10 VOORBEELD 4: DWARSREGELAAR

- Groot koppelnet
- Ongewenste vermogenstromen (“Loop flows”)
- Verbetering met geregelde fasedraaiende transformator (dwarsregeltransformator)
  - Tweemaal 1000 MVA
  - Regelbaar tussen  $-15$  en  $+15$  graden

## Voorbeeld 4: dwarsregelaar



**Mogelijkheden met macro's, voorbeeld 4: Regeling fasedraaiende transformator**

```

// Gewenste uitwisseling in Meeden (MW)
set(DwarsMW,0) //
////////////////////////////////////
set( ExportHoog , DwarsMW )
add( ExportHoog , 50 )
set( ExportLaag , DwarsMW )
subtract( ExportLaag , 50 )
// Initiële gegevens
text('Dwarsregelaar 1 fasedraaiing: ',transformer('Dwars1:MEE380-MEE380
dwars').PhaseShift,' graden')
text('Dwarsregelaar 2 fasedraaiing: ',transformer('Dwars2:MEE380-MEE380
dwars').PhaseShift,' graden')
loadflow(0,L,true)
if( Network.Result,=,'LF' )
// De export in Meeden is de som van P in de twee transformatorverbindingen
Set( ExportMeeden , transformer('Dwars1:MEE380-MEE380 dwars').P1)
Add( ExportMeeden , transformer('Dwars2:MEE380-MEE380 dwars').P1)
if ( ExportMeeden , > , 0 )
    text( 'Export Meeden : ' , ExportMeeden:7:2 , ' MW')
else
    text( 'Import Meeden : ' , Neg(ExportMeeden):7:2 , ' MW')
end
// Bekijk de export naar België
set ( ExportNLB , 0 )
forselection(line('NL-B'),MijnLijn)
    Add ( ExportNLB , MijnLijn.P2 )
end
text( 'Export NL -> B: ' , ExportNLB:7:2, ' MW')
end

// Begin van de regelacties
set( klaar , false )
text('=====')
text('Start regelactie:')
while( klaar , = , false )
    set( ExportMeeden , transformer('Dwars1:MEE380-MEE380 dwars').P1)
    add( ExportMeeden , transformer('Dwars2:MEE380-MEE380 dwars').P1)
    if( ExportMeeden , > , ExportHoog )
        if ( transformer('Dwars1:MEE380-MEE380 dwars').PhaseShift , < , 15 )
            text( 'Export in Meeden: ' , ExportMeeden:8:3, ' MW; fase neemt toe')
            add ( transformer('Dwars1:MEE380-MEE380 dwars').PhaseShift , 1 )
            add ( transformer('Dwars2:MEE380-MEE380 dwars').PhaseShift , 1 )
        else
            set( klaar , true )
        end
    end
    if( ExportMeeden , < , ExportLaag )
        if ( transformer('Dwars1:MEE380-MEE380 dwars').PhaseShift , > , -15 )
            text( 'Export in Meeden: ' , ExportMeeden:8:3, ' MW; fase neemt af')
            subtract ( transformer('Dwars1:MEE380-MEE380 dwars').PhaseShift , 1 )
            subtract ( transformer('Dwars2:MEE380-MEE380 dwars').PhaseShift , 1 )
        else
            set( klaar , true )
        end
    end
end

loadflow(0,L,true)

if(klaar , = , false)
    if( ExportMeeden , <= , ExportHoog )

```

```

        if( ExportMeeden , >= , ExportLaag )
            set( klaar , true )
        end
    end
end
end

text('=====')
text('Na regelactie:')
text('Dwarsregelaar 1 fasedraaiing: ',transformer('Dwars1:MEE380-MEE380
dwars').PhaseShift,' graden')
text('Dwarsregelaar 2 fasedraaiing: ',transformer('Dwars2:MEE380-MEE380
dwars').PhaseShift,' graden')
if( Network.Result,='LF' )
    Set( ExportMeeden , transformer('Dwars1:MEE380-MEE380 dwars').P1)
    Add( ExportMeeden , transformer('Dwars2:MEE380-MEE380 dwars').P1)
    if ( ExportMeeden , > , 0 )
        text( 'Export Meeden : ' , ExportMeeden:7:2 , ' MW')
    else
        text( 'Import Meeden : ' , Neg(ExportMeeden):7:2 , ' MW')
    end
    set ( ExportNLB , 0 )
    forselection(line('NL-B'),MijnLijn)
        Add ( ExportNLB , MijnLijn.P2 )
    end
    text( 'Export NL -> B: ' , ExportNLB:7:2, ' MW')
end

```

## 11 CONCLUSIE

- Macro's zijn onder andere bruikbaar voor:
  - Varianten-analyses
  - Zoeken naar mogelijke overbelasting
  - Studies met groeiende of wisselende belastingen
  - Specifieke regelingen
  - Simulaties
- Voordelen van macro's:
  - De mogelijkheden zijn onbeperkt
  - Besparen werk bij repeterende taken
  - Bieden uitgebreide rapportagemogelijkheden
- Voor een uitgebreide beschrijving, zie:
  - Vision handleiding
  - Macro user reference