

**Virtual Instrumentering:  
LabView Introduktion 3**

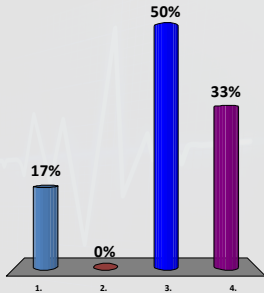
Samuel Alberg Thryssøe, PhD, PostDoc,  
Kontakt info: Email: sat@iha.dk, Tlf: +45 25533552



**Repetition fra sidst**

**Plotning af koord med uregelmæssig afstand**

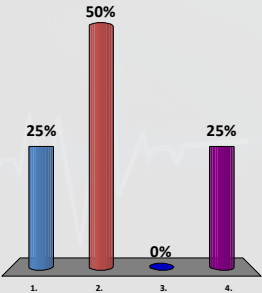
1. Waveform Chart
2. Waveform Graph
3. XY Graph
4. Indicator Graph



Category	Percentage
1	17%
2	0%
3	50%
4	33%

**Plotning af data med fastlagt startværdi og delta X**

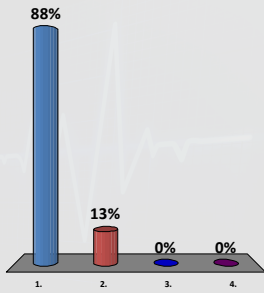
1. Waveform Chart
2. Waveform Graph
3. XY Graph
4. Indicator Graph



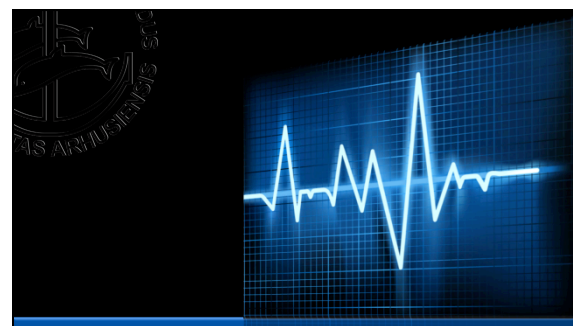
Category	Percentage
1	25%
2	50%
3	0%
4	25%

**Plotning af data med løbende opdatering**

1. Waveform Chart
2. Waveform Graph
3. XY Graph
4. Indicator Graph



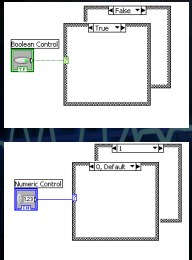
Category	Percentage
1	88%
2	13%
3	0%
4	0%



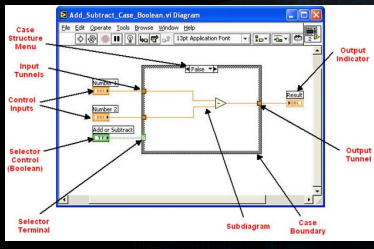
**Styring af program flow:  
Case strukturer**

## Case Strukturer

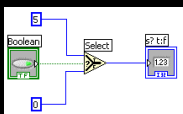
- Har en eller flere underdiagrammer (cases)
- Kun en vil køre når strukturen kaldes
- Den forbundne kontrol bestemmer hvilken case der køres
  - Kan være Boolsk, streng, integer, enum etc.
- Højreklik på struktur kanten for at tilføje/fjerne cases



## Case strukturer



## Select Funktion



- Returnerer T(rue) værdien, hvis det boolske input er True
- Returnerer F(false) værdien, hvis det boolske input er False

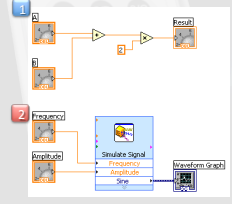
## 3.1 – Cases

- 3.1.1 - Temperatur Case
- 3.1.2 - Terningekast Respons
- 3.1.3 - Kvadratrod
- \*3.1.4 - Addition/Subtraktion
  - Stjernemarkede øvelser er valgfrie og tages, hvis de kan nås

## Styring af program flow: Select strukturer



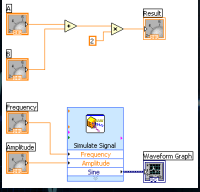
## Hvilken blok kører først?



Option	Percentage
1. Blok 1	22%
2. Blok 2	0%
3. De kører samtidig	78%

## Dataflow Programmering

- Block Diagram Udførelse
  - Kommer an på data flow
  - Eksekverer IKKE venstre til højre/top til bund
- En node eksekveres når data er tilstede på ALLE inputs
- Noder giver data videre til output terminaler når de er færdige

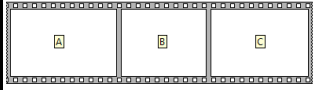


## Sekvens Strukturer

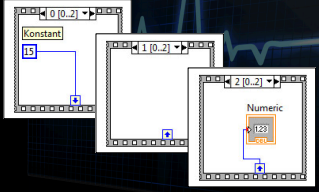
- Kontrollerer dataflow i VI block diagrammer
- Opdeler VI i frames
  - Ligner film-strimmel
- Flow kører fra frame til frame når alle noder i den enkelte frame er færdige
- Tegnes som loops og case strukturer
- Højreklik ramme for at tilføje/fjerne frames
  - To typer af sekvens strukturer:
    - Flat
      - Giver bedst overblik
    - Stacked
      - Ligner case strukturer
      - Giver mulighed for at bruge lokale sekvens variable

## Sekvens strukturer



Flad struktur



Stak struktur



## Lokale variable

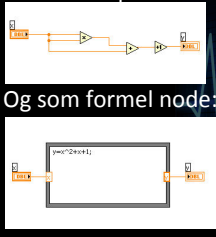
- Du kan læse fra / skrive til kontroller og indikatorer overalt i sekvens diagrammet
- Højreklik ikonet og vælg
  - Create → Local variable
  - Højreklik den lokale variabel for at skifte mellem
    - Read 
    - Write 

## 3.2 – Sekvensstrukturer

- 3.2.1 - Responstid
- \*3.2.2 - Tid til Match 2  
Valgfri øvelse – hvis der er tid

## Formel noder

- En formel node kan indeholde formler
- $Y=x^2+x+1$  på normal block diagram vis:
- Og som formel node:



### Oprettelse af formel node

### 3.3 – Formel Nodes

- 3.3.1 - Temp Konvertering 2

### Sub VIs

### Sub VIs

- Samling af operationer
- Et lav-niveau VI inden i et høj-niveau VI
  - Reducerer antallet af Block Diagram ikoner
  - Øger læsbarheden
  - Modulær opbygning
- Analoge til funktioner i tekst-baserede programmerings sprog

### LabView SubVIs ~ Funktioner

<p>Function Pseudo Code</p> <pre>function average (in1, in2, out) {   out = (in1 + in2)/2.0; }</pre>	<p>Calling Program Pseudo Code</p> <pre>main {   average (in1, in2, pointavg) }</pre>
<p>SubVI Block Diagram</p>	<p>Calling VI Block Diagram</p>

### Oprettelse af SubVIs

- 2 metoder
  - Lav et blankt VI og indsæt de ønskede funktioner
  - Marker en række funktioner i Block Diagrammet og vælg Edit -> Create SubVI
- Kontroller → Inputs
- Indikatorer → Outputs

## Oprettelse af SubVIs

- Marker området, der skal blive til et Sub-VI
- Vælg Edit → Create SubVI

## Ændring af Ikon

- Vigtig for at kunne skelne SubVIs
- Højreklik standard-ikonet øverst th på Block Diagrammet og vælg: Edit Icon

## Ændring af forbindelser

- Højreklik VI ikonet øverst th på Front Panelet og vælg: Show Connector
- Ikonet skifter til connector
  - Rediger terminaler ved at højreklikke på connectoren og vælg Add/Remove Terminal
  - Eller vælg standard konfigurationer
    - Højreklik Connector → Patterns
- Forbind inputs (tv) med kontroller
- Forbind outputs (th) med indikatorer

## Ændring af forbindelser

## Forbind SubVIs i Block Diagrammet

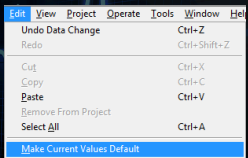
- Terminalerne bliver navngivet efter navnene på kontroller/indikatorer i SubVIs

## Forbindelses egenskaber

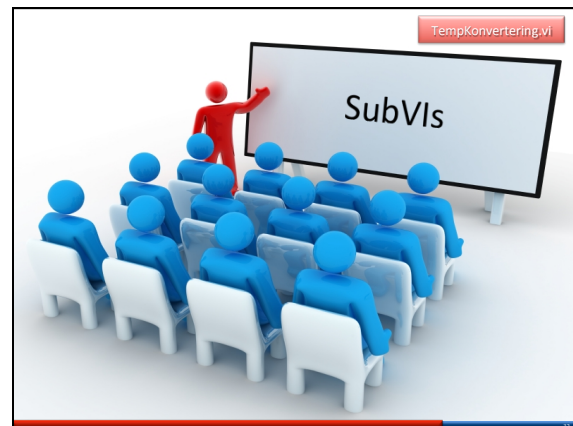
- Forbindelser kan sættes til at være
  - Required
  - Recommended
  - Optional

## Default værdier

- Værdierne i kontroller kan sættes til at være standard
- Edit -> Make Current Values Default
- Benyttes, hvis input ikke er forbundet



Edit	View	Project	Operate	Tools	Window	Help
Undo Data Change						Ctrl-Z
Redo						Ctrl-Shift-Z
Cut						Ctrl-X
Copy						Ctrl-C
Paste						Ctrl-V
Remove From Project						
Select All						Ctrl-A
<b>Make Current Values Default</b>						



## 3.4 – SubVIs

- 3.4.1 - Vælg kolonne
- 3.4.2 - Terningekast 3\_SubVIs
- \*3.4.3 - Random LED\_SubVI  
Valgfri øvelse – hvis der er tid

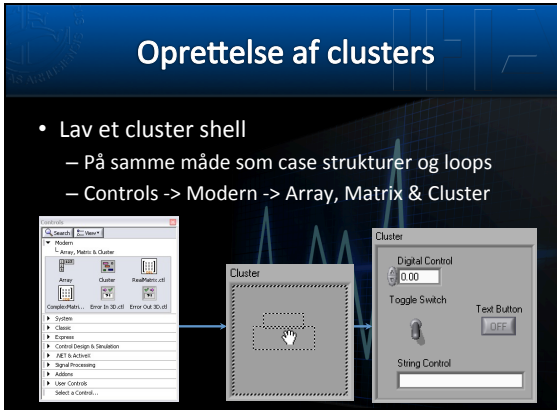


## LabView Clusters

- Analoge til Structs i C++ / Matlab
- Kombinerer en/flere data typer i en enkelt struktur
  - Fx boolean, strenge og integers
- Reducerer kompleksiteten af block diagrammer
- Unbundle giver adgang til de enkelte elementer

## Oprettelse af clusters

- Lav et cluster shell
  - På samme måde som case strukturer og loops
  - Controls -> Modern -> Array, Matrix & Cluster



## Cluster Rækkefølge

- Ændring af rækkefølgen af cluster komponenter:
  - Højreklik kanten af clustret på front panelet
  - Vælg "Reorder Controls In Cluster..."
- To kasser kommer frem på alle cluster komponenter
  - Hvid = Nuværende plads
  - Sort = Ny plads
- For at flytte en komponent:
  - Vælg pladsen øverst
  - Klik på kontrollen



## Bundle / Unbundle

Terminal labels afspejler data typen

### Bundle By Name

input cluster → element 0 → name 0 → output cluster  
 element m-1 → name m-1 → element m-1

Opdaterer specifikke cluster objekt værdier

### Unbundle By Name

cluster of named → element 0  
 element m-1

Returnerer de cluster elementer, hvis navne du specificerer

## Cluster <-> Array

- Du kan skifte mellem clusters og arrays hvis
  - Alle cluster komponenterne har samme data type
  - Alle er boolske/numeriske/stringe etc.

## 3.5 – Clusters

- 3.5.1 - Cluster IO
- 3.5.2 - Cluster Bundling
- \*3.5.3 – Login  
Valgfri – hvis der er tid

