Welcome to the CoDeSys training module for the DCS800, ABB DC Drives.
If you need help navigating this module, please click the Help button in the top right-hand corner. To view
the presenter notes as text, please click the Notes button in the bottom right corner.

Note:
This module is an exercise without a speaker!
After completing this module, you will be able to

- Implement a visualization
- Connect and control the application visual

After completing of this module, you will be able to implement a visualization and connect and control the application visual.
The steps to create the exercise are the following.
Create a new project in CoDeSys and insert needed function blocks for communication with the drive. Accordingly define global variables for program communication and create the visualization window. Last step is to test the program.
First it is important to create a new POU from type program in programming language “Function Block Diagram”. After this configure the task configuration, please.
Now we can create the program in the several networks like in the picture on the right side. Network 1 includes a parameter read block which takes data from parameter 1.04, the speed actual value. In network 2 you can see that the value from parameter 8.01, the main status word, will be written to the variable “Main Status Word”. A value coming from CoDeSys should be written to parameter 23.01 using the block in network 3. This is the speed reference for the drive.
For an easy communication with the visualization it is necessary to use global variables. This variables can communicate with other POUs and the visualization tool, too. Define all variables like in the pictures, please.
Now it’s time to implement the visualization.

- Change to the visualization window.
- Click with right mouse button to “Visualizations” and select “Add Object”.
- Define a name for the new visualization.
- Then you get the new object DCS800Visu.
- The right window is the *editor window* for the several functions.
Now CoDeSys is configured to edit the visualization:

- Above the editor window you find the icon bar with all possibilities to edit the visualization.
- With this icons you can build the visualization.
- The following tools are possible: e.g., Button, Rectangle, Ellipse, Table, Trend line, Meter
Now we start a visualization example:

- For the example we need the tools like in the window.
- The meter should show the actual speed of the drive. It is connected to variable `SpeedAct`.
- The ellipses should show the state of parameter 8.01. It is connected to variable `MainStatWord`.
- In rectangle it is possible to get a speed reference.
Select the meter and place it at the **editor window**.

Now the meter must be configured.

Click to **Variable/Scale** and set the parameters like in the window.

In this window the following fields can be configured:

- Minimum / Maximum
- Main and Sub scale
- Unit
- Connection to **SpeedAct**
Next step is to define color areas

- Click to **Color Areas**
- Type in a begin and an end point of the area
- Click to **Color selection** and choose a color
- Then click to Add and the area will be saved
- In the same way the other areas must be configured
- This is the complete configuration

The picture on the right side shows you now the complete configuration.
The final product is like in the picture.
Next visualization tool is the ellipse.
Click to icon *ellipse* and draw 4 pieces.
A double click to an ellipse opens the configuration window.

- Change to **Variables**
- Type in the variable for changing color
  - Press F2
  - MainStatWord

- The final product is the picture on the right side.
- Next visualization tool is the ellipse.
- Click to icon ellipse and draw 4 pieces.
- A double click to an ellipse opens the configuration window.
- Now change to variables
- Type in the variable for changing color
• Change to field “color” click to alarm color – inside and select a color from the list.
• Now if everything is correct, the color change if the selected bit in MainStatusWord change the state.
• In the same way configure the other ellipses.
• The complete editor window looks like this picture.
• Next step is to create a rectangle to type in a speed reference.
• Click to icon rectangle.
• Draw a rectangle into "editor window".
• A double click to rectangle opens the configuration window.
• Now click to text and type in the display format.
Change to variables and mark the field "Textdisplay".
Press F2 and select variable "SpeedRef" from the list.
Change to item "input" and set a tick into field "Text Input".
Select Keypad and define minimum and maximum limitations.
The application function is like the following description:

- The meter shows actual speed of the motor
- The 4 ellipses show the state of the drive, e.g.
  - 8.01 bit 0 (RdyOn)
  - 8.01 bit 1 (RdyRun)
  - 8.01 bit 2 (RdyRef)
  - 8.01 bit 7 (Alarm)
- The rectangle is used to type in a speed reference
  - Minimum –1500 rpm
  - Maximum +1500 rpm
If the program is correct, it can be tested:

- Build the program.
- Login and click Run.
- Start the drive and check the actual speed on the meter.
- Set parameter 11.03 using DriveWindow light to "SpeedRef" 23.01.
- Type in a speed reference between −1500…+1500 rpm into CoDeSys visualization.
• If you will learn more about CoDeSys application programming, use a course on ABB Academy, please.
Additional information

- Links to related information
  - 3S-software.com
  - DC-Drive-News (Intranet)

- Additional references
  - Application Manual (3ADW000 199)
  - Firmware Manual (3ADW000 193)
  - Hardware Manual (3ADW000 194)
  - Training Material
Glossary

- **CoDeSys**
  Controller Development System (software tool)

- **Memory Card**
  Flash memory

- **DriveWindow Light**
  Software Tool for commissioning and maintenance using AC/DC

- **Target**
  Interface between Drive and CoDeSys tool

- **Control Builder**
  Whole system with software and hardware

- **PLC_PRG**
  Main program which is used in all applications

- **POU**
  Program Organization Unit

- **Library**
  It includes function blocks which are given or designed by other users
Thank you for your attention. You may now go ahead and move on to the next unit.