Welcome to the [Module title] training module for the [Product name], ABB [Product family].
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.
After completing this module, you will be able to

- Generate user events

After completing of this module, you will be able

- To generate user events
An event is a message or a problem report. The event is shown in DriveWindow, DriveWindow light and on the DCS800 panel. There are 16 Faults, 16 Alarms and 16 Notices available. The definition of this events is happened in CoDeSys.

“User Faults” will be handled like fault level 1 and shuts down the system (Firmware-Manual: Fault Tracing)

The text for all events can be defined by the user.
Let's do a little exercise. We would like to control the actual speed of a motor. At 85% of nominal motor speed there should be generated an alarm and with 95% a fault. The application should have the functionality for both motor directions.

In this application we work with internal values of speed measurement. So, we must scale the physical speed from 1500 rpm into internal scaling. The speed reference is given with analog input 1.
First start CoDeSys and create a new project. We use programming language “CFC” and the program name “PLC_PRG”.

If the settings are done correctly, insert the boxes like in the picture and connect the function blocks.
Now we need several texts for the alarms and faults. The event messages are saved in parameters which are defined in the “Parameter Manager”. We use function block “Event Init” to allocate the texts for the right event. In this exercise we are saving the fault messages in user parameter 60.01 and the alarms in 60.02.

Then set the communication and task parameters to get a working communication between drive and pc.
Now we change to the “Parameter Manager” and define new user parameters for the message texts. Field Variable is the interface between CoDeSys and the parameter. It must be a global variable which has a dot before the variable. Data type of this parameters are an “Enumeration”. What you can see later in the parameter list is the name of the new parameter group and the name of the parameter index which is defined in field “Name”.

Attention: The fields Def, Min, Max cannot be set now. This has to be done later!

Don’t forget the dot, because of Global Variables!
The slide before shows that we need global variables to connect CoDeSys with the parameters. So, we must define 2 global ones.

Next step is to define the alarm and fault texts. This can be done in the window “Data Types”. Add a new data type and select a name for the new data type. In this exercise we define the names Alarm and Fault.

Now the 2 new data types are added to the list!
This slide shows how we add texts to the several data types. The message text is in brackets split with a comma. First one in the bracket is allocated to the smallest event number. See also the picture on the right side.

There are in maximum 12 characters allowed!

<table>
<thead>
<tr>
<th>Create data types</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create the text for the messages</td>
</tr>
<tr>
<td>→ Delete the structure (STRUCT ... END_STRUCT) and set the type</td>
</tr>
<tr>
<td>The texts (max. 12 characters) in the brackets will be shown in DW, DWL and on the DCS800 panel</td>
</tr>
<tr>
<td>- Alarm: CriticalValP  → A310</td>
</tr>
<tr>
<td>- CriticalValN  → A311</td>
</tr>
<tr>
<td>- Fault: OverSpeedPos  → F610</td>
</tr>
<tr>
<td>- OverSpeedNeg  → F611</td>
</tr>
<tr>
<td>Repeat the procedure for other objects (Faults, Alarms, Notices)</td>
</tr>
</tbody>
</table>

**ENUM (enumerated data type):**

**ENUM is a STRING-Object, which is used for listings**
Now the event messages are defined, and we can set the values in the “Parameter Manager”. Normally, we set the first text in all fields because we don't need a selection for this parameter. It's only to store data.

If the program is ready, we can download it into DCS800 and test the functionality. The 2 new parameters will be shown in the parameter list with the default text.
Before we can do a test with this new application, it necessary to configure the drive first. The easiest way to check the application is to play with the unit in local mode with the switches.

Start the drive and change the speed reference up and down. If the drive is faster than 1275 rpm you get an alarm message. A motor speed faster than 1425 rpm generate a fault and the drive switch off.
Key points of this module was to create a little application with user events.
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 (eLearning)
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

ABB
Thank you for your attention. You may now go ahead and move on to the next unit.