Automating Hydroelectric Power Plant

By Step 7 V15.1 WinCC RT Advanced

Introduction:

The idea here is to program controls for a very CRUDE hydroelectric power plant. I start with a base program that simulates a changing environment and manages our Analog input signals accordingly. My job is to write a program that incorporates these signals and controls various Analog and digital devices in order to run the system with a safe and stable process. We also want to create an HMI that will allow the Operator to interact easily and effectively with our system.

Project Description:

The project is divided mainly in two parts one is programming the PLC and the other is to design or create an HMI for the above-mentioned project. Firstly, we will deal with programming of PLC and in this I have programmed the project in four different programming languages which includes LAD, FBD, STL and SCL. I have created my program in such a way that any language can be chosen at any time which will provide the same result.

I have programmed my project in two different sections and they are:

1. Main Organization Block (main OB) which includes all different functional blocks and functions they are:

   1.1 Sequence of Operations in Sequential flow chart (SFC) language.

   1.2 Configuration of IO - includes all Analog input/out signals and digital input/output signals.

   1.3 Alarm Conditions - includes seven different conditions like Emergency Stop, Low oil flow, High oil flow, Overcurrent, Overspeed, High oil temperature, and Brake failure.

   1.4 HOA's (Hand on, Off, and Auto operation) - includes Oil pump VFD, Generator Brake, Station-Interlock, Baffle, and Alarm Horn.

   1.5 Program Modes - includes Warmup mode, Stabilize mode, Generation mode, Cooldown mode, Idle mode, and Fault mode.

   1.6 Hour Meters - includes Hour meter minutes, Hour meter hours, and Hour meter days.

2. PID controller Organization Block (OB) which is also a cyclic interrupt block and includes:

   2.1 Oil high flow alarm

   2.2 Setpoint conversion

   2.3 HOA's for Baffle and Oil pump VFD

   2.4 PID ON handling

   2.5 PID OFF handling

Now, designing of HMI I choose WinCC RT Advanced (HMI) in which I have create four different screens with navigation i.e. any screen can be selected any time from the previous screen and also exit app switch which will stop runtime of HMI.

The four different screens are:

1. System Status - Show system mode, alarm indicators, important process values, high-level system controls (like a start and stop button).
2. System Configuration - Give the Operator controls over all devices to include manual setpoints for Analog devices. Also display the status / values of inputs signals. You may also want to show alarm indicators (not notification bits) for any devices that may be inhibited.

3. Alarm Management - Show alarm history, alarm reset and silence buttons, and indicators which show the presence of alarm bits (just notifications).

4. System Overview - Show a graphical representation of our system with relevant process values appearing at their points of collection. You can decide where exactly those may be when not clear.

Technical Specification:

1. IO Configuration –

1. Baffle:

Signals: 1 DQ with tag name “Baffle_En” and 1 AQ with tag name “Baffle_Out”
Output scaling value range: 0-27648
HMI Display values: 0-100% open
Polarity: Reverse (higher output signal value closes the baffle more)

2. Rotor Speed:

Signals: 1 AI with tag name “RPM_In”
Input scaling value range: 0-27648
HMI Display values: 0-200 RPM
Polarity: Straight (higher input signal value means higher RPM)

3. Oil Pump VFD:

Signals: 1 DQ with tag name “OilPumpVFDen_Out” and 1 AQwith tag name “OilPumpVFDsp_Out”
Output scaling value range: 0-27648
HMI Display values: 0-100%
Polarity: Straight

4. Oil Flow:
Signals: 1 AI with tag name “OilFlow_In”
Input scaling value range: 0-27648
HMI Display values: 0-30 GPM
Polarity: Straight

5. Oil Temperature:
Signals: 1 AI with tag name “OilTemp_In”
Input scaling value range: 0-27648
HMI Display values: 0-500 degrees
Polarity: Straight

6. Power Station Interlock:
Signals: 1 DQ with tag name “StationInterlock_Out”
Polarity: Straight

7. Alarm Horn:
Signals: 1 DQ with tag name “AlarmHorn_Out”
Polarity: Straight

8. Generator Brake:
Signals: 1 DQ with tag name “Brake_Out”
Polarity: Straight

9. E-Stop:
Signals: 1 DI with tag name “E-Stop”
Polarity: Reverse
10. AC Current:
Input scaling value range: 0-27648
HMI Display values: 0-700 kW
Polarity: Straight

2. Sequence of Modes:

1. Warmup:
When Start pushbutton is pressed, the sequence begins with this step. Allow the rotor to turn at a speed somewhere between 20 and 50 RPMs and run the oil pump at some low output level (2-10 GPM) until the oil temp reaches 150 degrees. Once we’ve maintained at least 150 degrees for 10 seconds, we want to transition into the next mode.

2. Stabilize:
Generator should speed up until desired kW output is attained. Once we’re stable within 5kW (+/-) of the setpoint for at least 10 seconds, we should transition into the next mode.

3. Generation:
Keep everything running and close the interlock.

4. Cooldown:
Once the Stop button has been pushed, open the interlock, slow the rotor to between 20 and 50 RPMs and maintain a high oil flow until the oil temp is below 150 degrees. Once we’re below for ten seconds, we transition into Idle mode.

5. Idle:
Interlock and baffle are deenergized and brake is engaged

6. Fault:
When we hit an alarm, the sequence should be aborted according to the alarm details, and we should go into Fault mode (which is not part of our sequence).

Conclusion:

With my prior knowledge in TIA portal step 7 WinCC RT Advanced, I was able to program the PLC in for different languages and also designed the HMI for the visual analysis of all the different components of the power plant. The power plant can be operated in manual as well as in auto mode depending on the requirement. Finally, the Power plant project was programmed and tested successfully.