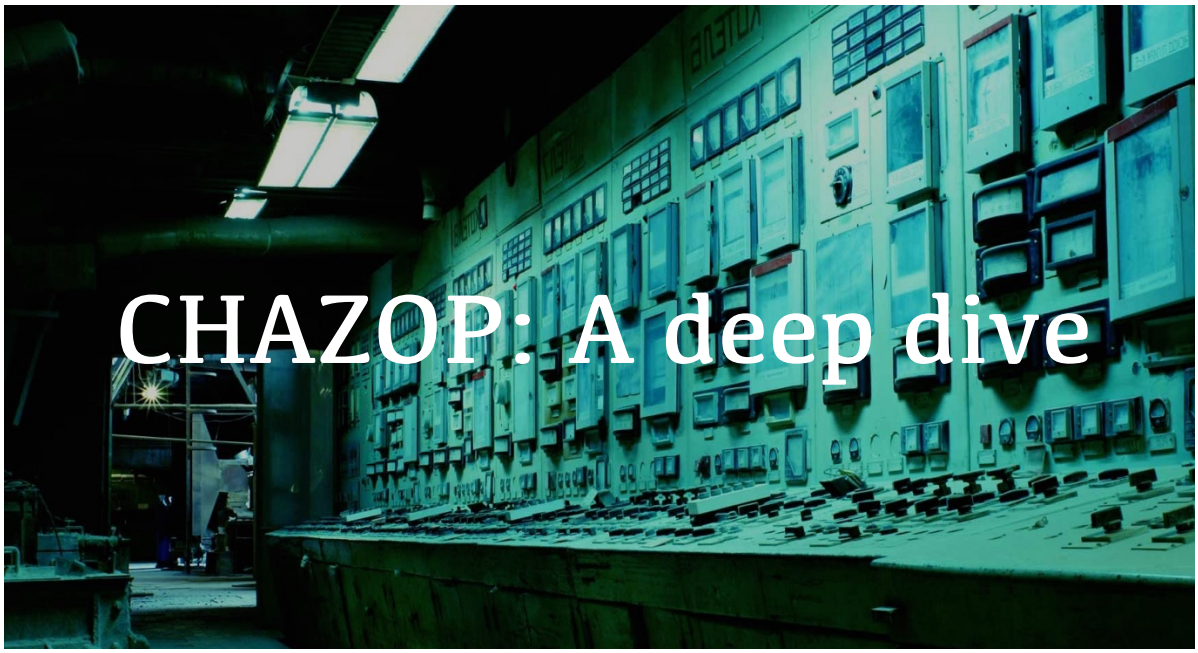

WHAT THE CHAZOP COURSE COVERS

MODULE 1: Concept of CHAZOP

- Objective and Scope of CHAZOP
- Information provided by CHAZOP
- What types of control system risk analysis are available?

MODULE 2: CHAZOP Input Data and Procedure

- Typical Input Data problems for conducting CHAZOP
- Pre-screening of HAZOP report
- CHAZOP Procedure
- CHAZOP Methodology for continuous operations
- CHAZOP Methodology for batch or sequence operations
- Roles of chairman and scribe

MODULE 3: CHAZOP Methodology, Execution and Recording of the Study

- Executing a CHAZOP workshop: functional areas and failure classes
- Risk ranking
- Recording the CHAZOP study
- Reporting the CHAZOP study

MODULE 4: CHAZOP example in detail

- Detailed checklists: continuous and batch mode CHAZOP
- Example: using input data to generate the CHAZOP worksheet

INTRODUCTION TO MODULE 1:
Concepts of CHAZOP

CHAZOP is a Control systems HAZOP/ Computer HAZOP study, for carrying out the safety and reliability analysis of existing or planned control and computer systems. The objective of CHAZOP is to control the system risk if any of the following hold true for the project:

- The control system is not trivially simple
- The control system is programmable
- The control system is customised, unusual or novel in design
- The operators' experience with the control system is limited
- The system is interconnected with other systems and networks
- The maximum severity of potential upset consequences is significant, as assessed during HAZOP or QRA study
- The application software (the code that implements the specific control and safety functions in the project) is specially written, and not directly copied from an existing project.

With CHAZOP we review how a control and computer system can fail and deviate from design. We also review the consequences of the deviations, and whether current safeguards are adequate to prevent the consequences or whether design changes are required.

To understand how and when to use it, we need to know the background issues: What is the objective and scope of CHAZOP? Why are we performing CHAZOP? What information and what types of control system risk analysis are available, and which is the best analysis tool for each situation? These topics, which are fundamental to a correct application of CHAZOP, are covered in this module.

Terms and abbreviations

Term	Definition
Checklist	A form of PHA more suited for analysing well-defined situations such as facility siting
Consequence	Unwanted result of the harmful event resulting from a hazard
Control system hazard and operability study (CHAZOP)	A form of workshop study PHA that systematically examines upsets within a control system that will affect a process
Harm	Undesired negative impact on risk receptors from a hazardous incident – for instance, an injury or fatality
Hazard	Property or action that has the potential to cause harm under certain conditions

1. Objective of CHAZOP

CHAZOP is one of a suite of methods for assessing the safety and operability of a process plant, and the maintainability of the process's control system. It is applicable to facilities such as oil and gas, petrochemical, general chemical, pharmaceutical, power and energy, and nuclear. A similar approach can also be applied in other applications such as machine control systems.

The purpose of CHAZOP is to find possible causes of process upset due to control system failure. The study identifies credible consequences arising from these causes, which may be

- Immediate, e.g. plant operating in unsafe conditions leading to injury, environmental impact, equipment damage and downtime
- Delayed, e.g. loss of safeguards or redundancy, increased likelihood of future failures, or a potential cyber security incident

Existing safeguards against these consequences are assessed to determine whether they are sufficient.

What is a control system?

Most plants use some variation of computer-based control systems to run and protect themselves. These contain a processor connected by input/output (I/O) ports to external sensors and process control devices such as temperature and pressure sensors, valves and motor controllers for pumps, fans and compressors.

Control systems are known by various names such as Basic Process Control Systems (BPCS), Distributed Control Systems (DCS), Programmable Logic Controllers (PLC), Supervisory Control and Data Acquisition Systems (SCADA), and Safety Instrumented Systems (SIS).

Failure of an element of a control system element may cause unpredictable operating conditions, hardware inputs and outputs frozen or in unpredictable states, or even put the process in a dangerous condition leading to an accident. Operators may not be able to make changes or activate/deactivate overrides or bypasses and may be unable to turn equipment on or off. Control system issues could also lead to cyber security breaches.

Continuous vs. batch mode CHAZOP

Continuous CHAZOP	Batch mode CHAZOP
Follows wide-ranging checklist	Focuses on deviation from sequence intent
Limited attention to human factors	Substantial attention to human factors (unless sequence is fully automatic)
Identifies failure causes and consequences	Focuses on identifying causes only. These can then be fed into batch mode HAZOP
Best done after HAZOP	Best done before HAZOP
Needs interdisciplinary group	Can be done by small group from process and C&I depts

Question 1.2

During batch mode CHAZOP, why isn't it necessary to consider every step in the sequence one by one?

Go to suggested answer.

4. Software Integrity and Criticality Analysis

Software faults leading to unforeseen behaviour are a significant cause of control system failure. Hardware faults can occur randomly, whereas software faults may be pre-existent in the system, and may remain hidden until revealed by a specific combination of conditions.

The approach to **software integrity analysis** is therefore focused on assessing the quality of software design and testing—addressing potential faults before they are revealed during live operations.

We need to consider 2 main types of software:

- Embedded software – the software environment running in the control system. This consists of the operating system (e.g. Windows NT), and the control system manufacturer's software that implements all of the control functions (e.g. Emerson's DeltaV). Embedded software is essentially the same for every user and application.
- Application software – the specific code (e.g. function blocks, ladder logic) running inside the embedded software, which implements specific functions for each process. Application software is unique to each process and plant.

Also, consider how software faults can be managed. It is impossible to prove software is error-free by testing, as the number of possible tests would be unfeasibly large. So we rely on 2 main approaches to justify a claim that the software is sufficiently error-free:

