|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **TSC Category** | Development and Implementation | | | | | |
| **TSC Title** | Control System Programming | | | | | |
| **TSC Description** | Develop capabilities in areas of communications and remote operations by programming logic circuits and erasable programmable read-only memory for ships, rigs and/or conversions | | | | | |
| **TSC Proficiency Description** | **Level 1** | **Level 2** | **Level 3** | **Level 4** | **Level 5** | **Level 6** |
|  | **ICT-EPM-2036-1.1** | **ICT-EPM-3036-1.1** | **ICT-EPM-4036-1.1** |  |  |
|  | Apply basic hardware programming techniques to build peripheral systems around the programmable logic controllers (PLC) and troubleshoot programming errors in the codes | Implement hardware programming techniques to enhance functionality of equipment and systems by using appropriate process parameter measuring devices and utilising their outputs to control operations | Develop programmable control systems by incorporating new technologies and linking them to operating principles of equipment and systems on-site and advise involved parties on programming techniques |  |  |
| **Knowledge** |  | * Fundamental concepts of programming * Logic arguments in programming * Standard built-in functions and sub-routines | * Procedures for hardware programming in the areas of communication, remote operation and sensor capabilities * Basic principles of mechanical engineering * Principles of interaction between electronic and mechanical components * Best practices and industry innovations in the field of coding and programming * Types of communication and remote or autonomous operation sensors and feedback units | * Advanced principles of mechanical engineering * Advanced concepts of communication systems * Advanced principles of sensor technologies * Types of fully automatic, semi-automatic and manual control systems * Logic flow for implementation of systems for ships, rigs and/or conversions |  |  |
| **Abilities** |  | * Apply object-oriented programming * Construct programmes with user-defined functions and sub-routines * Convert algorithms into programming codes | * Infer final mechanical effects of sensor programming * Identify potential hazards in using programmable sensors in lieu of manpower * Create contingency procedures for local and remote interventions | * Design control systems for a variety of applications * Review programming inputs and guide improvements * Conduct feasibility studies for new programmable equipment to ensure profitability * Predict aftersales support requirements |  |  |
| **Range of Application** |  | | | | | |