**DAILY ASSESSMENT FORMAT**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | **30/may/2020** | **Name:** | **Prashantha naik** |
| **Course:** | **Logic design** | **USN:** | **4al17ec074** |
| **Topic:** | **Applications of Programmable logic controllers:** | **Semester & Section:** | **6th b** |
| **Github Repository:** | **prashanth\_course** |  |  |

|  |
| --- |
| **FORENOON SESSION DETAILS** |
| **Image of session** |
| **Report – Report can be typed or hand written for up to two pages.**  Programmable logic controllers [PLC] are computer-based, solid-state, single processor devices that emulate the behavior of an electric ladder diagram. capable of controlling many types of industrial equipment and entire automated systems. The term logic is used because the programming is primarily concerned with implementing logic and switching operations. Input devices such as switches, and output devices such as motors, being controlled are connected to the PLC and then the controller monitors the inputs and outputs according to the machine or process. Originally PLCs were designed as a replacement for hard-wired relay and timer logic control systems. (Hard-wiring means that all of the components were manually connected by wires). PLC consists of two parts i.e. the PLC hardware and programming.  The [PLC](https://www.iaasiaonline.com/what-a-plc-upgrade-can-do-for-you-2/) receives information from connected sensors or input devices, processes the data, and triggers outputs based on pre-programmed parameters.  Depending on the inputs and outputs, a PLC can monitor and record run-time data such as machine productivity or operating temperature, automatically start and stop processes, generate alarms if a machine malfunctions, and more. PLCs are a flexible and robust control solution, adaptable to almost any application.  There are several key features that set PLCs apart from industrial PCs, microcontrollers, and other industrial control solutions:  **I/O** – The PLC’s CPU stores and processes program data, but input and output modules connect the PLC to the rest of the machine; these I/O modules are what provide information to the CPU and trigger specific results. I/O can be either analogue or digital; input devices might include sensors, switches, and meters, while outputs might include relays, lights, valves, and drives. Users can mix and match a PLC’s I/O in order to get the right configuration for their application.  **Communications** – In addition to input and output devices, a PLC might also need to connect with other kinds of systems; for example, users might want to export application data recorded by the PLC to a supervisory control and data acquisition (SCADA) system, which monitors multiple connected devices. PLCs offer a range of ports and communication protocols to ensure that the PLC can communicate with these other systems.  **Human Machine Interface (HMI)**– In order to interact with the PLC in real time, users need an HMI. These operator interfaces can be simple displays, with a text-readout and keypad, or large touchscreen panels more similar to consumer electronics, but either way, they enable users to review and input information to the PLC in real time.  PLCs are used for continuously monitoring the input values from sensors and produces the outputs for the operation of actuators based on the program. Every PLC system comprises these three modules:  ***CPU Module***  A CPU module consists of central processor and its memory. The processor is responsible for performing all the necessary computations and processing of data by accepting the inputs and producing the appropriate outputs.  ***Power Supply Module***  This module supplies the required power to the whole system by converting the available AC power to DC power required for the CPU and I/O modules. The 5V DC output drives the computer circuitry.  ***I/O Modules***  The input and out modules of the programmable logic controller are used to connect the sensors and actuators to the system to sense the various parameters such as temperature, pressure and flow, etc. These I/O modules are of two types: digital or analogue.  ***Communication Interface Modules***  These are intelligent I/O modules which transfers the information between a CPU and communication network. These communication modules are used for communicating with other PLC’s and computers, which are placed at remote place or far-off locate.  The program in the CPU of programmable logic controller consists of operating system and user programs. The purpose of the operating system with CPU is to deal with the tasks and operations of the PLC such as starting and stopping operations, storage area and communication management, etc. A user program is used by the user for finishing and controlling the tasks in automation.  **Programming A PLC**  In these modern times, a PC with specially dedicated software from the PLC manufacturer is used to program a PLC.  The most widely used form of programming is called ladder logic. Ladder logic uses symbols, instead of words, to emulate the real world relay logic control. These symbols are interconnected by lines to indicate the flow of current through relay like contacts and coils. Over the years the number of symbols has increased to provide a high level of functionality.  The completed program looks like a ladder but in actuality it represents an electrical circuit. The left and right rails indicate the positive and ground of a power supply. The rungs represent the wiring between the different components which in the case of a PLC are all in the virtual world of the CPU. So if you can understand how basic electrical circuits work then you can understand ladder logic.  Today, a number of different programming languages are used, but each PLC supplier has their own programming specifications based on the IEC 61131-3 standard. Although they have roughly the same sort of components found in many other computer systems, PLCs operate quite differently. A PLC operating cycle, or scan, consists of:   * Reading and storing the current value of each input, * Changing all physical outputs to match the output table values stored in data memory, * Sequentially executing the instructions in program memory, while storing any updated variables or outputs to data memory. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Date:** | **30/may/2020** | **Name:** | **Prashantha naik** | |
| **Course:** | **Python** | **USN:** | **4al17ec074** | |
| **Topic:** | **Python for Image and Video Processing with OpenCV** | **Semester&Section:** | **6th b** | |
| **Git hub repository** | **prashanth\_couse** |  |  | |
| **AFTERNOON SESSION DETAILS** | | | |
| **Image of session** | | | |
| **Report – Report can be typed or hand written for up to two pages.**  **Face detection**    **Vedio capturing** | | | |