The general working principle of a PLC

It stands for Programmable logic controller

Before PLC's the industry used hardwired panels which were very time consuming to wire, debug and change.

# Advantages of PLC

* Flexible and robust
* Faster response time
* Less and simpler wiring
* Solid-state - no moving parts
* Modular design - easy to repair and expand
* Handles much more complicated systems
* Sophisticated instruction sets available
* Allows for diagnostics “easy to troubleshoot”
* Cheaper
* The PLC can take both analog and digital inputs
* PLC's can handle a wide range of communication protocols

# A PLC consists of

* A processor unit (CPU) which interprets inputs, executes the control program stored in memory and sends output signals,
* A power supply unit which converts AC voltage to DC,
* A memory unit storing data from inputs and program to be executed by the processor,
* An input and output interface, where the controller receives and sends data from/to external devices,
* A communications interface to receive and transmit data on communication networks from/to remote PLCs.

It’s used in industrial applications for automation. A PLC can receive information from a connected sensor or another device. Once the data is received it can process the data and trigger pre-programmed instructions depending on the output. PLC's can also be connected to other systems to record data etc.

To interface with a PLC in real time an HMI is needed. HMI meaning a human machine interface these can vary greatly.

PLC's is an important part of the IoT world. They can connect to SQL databases to store data, or connect to the cloud in other ways with MQTT etc. A PLC program is usually written on a computer and then is downloaded to the controller

Most PLC programming software offers programming in Ladder Logic, or “C”. Ladder Logic is the traditional programming language. It mimics circuit diagrams with “rungs” of logic read left to right. Each rung represents a specific action controlled by the PLC, starting with an input or series of inputs (contacts) that result in an output (coil). Because of its visual nature, Ladder Logic can be easier to implement than many other programming languages. “C” programming is a more recent innovation. Some PLC manufacturers supply control programming software.

A PLC has to be made very robust; it should be able to survive in harsh environments for like 10 years. A PLC is often modular, it usually comes with a processor and a power supply and then you just add on so called "cards" that adds different functionality like analog / digital - in/out.

This means that the factory can add only the required functionality and if anything breaks, it can be replaced without having to rebuild the whole system. A PLC has optical isolation, because most industrial devices run on 12v or 25v, which is way too much for the microchip inside of the PLC. Optical isolation means that the data is transferred using lights and infrared sensors.

A PLC has to have a hard real time clock, meaning it will be able to run hard RTS. (RTS = real time systems). When running hard RTS like a PLC it will simple just cut off processes that would take too much power if ever stressed. Downside of running RTS is you can only run smaller programs.

PLC program scan cycle = PLC programs runs on cycles, and you have to fit your program into a cycle otherwise it'll be cut off. Alternatively, if it’s a bigger operation you will have to save to program and load it in the next cycle.

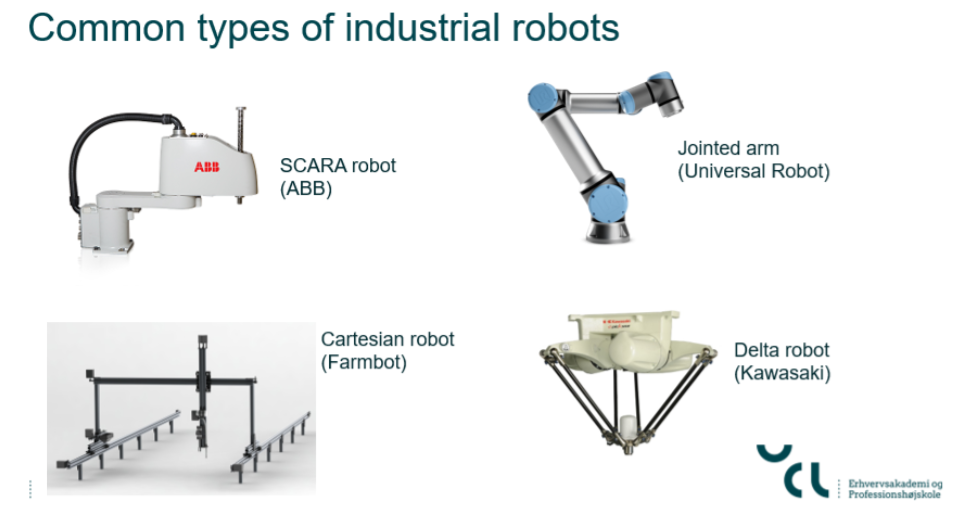
A cycle typically looks like this:

🡪 start 🡪 input scan 🡪 program scan 🡪 output scan 🡪 clean up 🡪

# PLC programming intro:

* Ladder programming language:
* Made for running in hard RTM systems
* Made by electricians, uses basic logic
* Coils (represented by a circle with holes in the middle) = output
* Two straight bars symbol = input (called contacts)
* You can also get inverted switches

Primary Source: <http://www.ieec.uned.es/investigacion/Dipseil/PAC/archivos/introtoplcs_SUPER.pdf>



SCARA robots, based on the robots’ axes of movement, are more modern compared to pick and place industrial robots. Both may serve the same function, but SCARA robots easily edge out the other resoundingly.

A SCARA robot consists of a base which is connected to the robot arm in a parallel-axis layout. The rotary joints provide compliance in the XY plane. In other words, the arm of a SCARA robot can move from left to right as well as back and forth. It, however, is rigid in the Z-axis such that it can only rotate on a theta-axis.

The design provides 4 degrees of freedom. Further, the arm can be folded back or outstretched. It is a feature that comes in handy depending on what length is necessary for the task it is being used for.

In regard to software, SCARA robots are designed to automate tasks but under the controller’s instruction. The controller interface is easy to use, and they tend to be responsive to commands.

Given that they come as a ready-made solution there is little if any additional programming required.

## Pros and Cons

Pros:

* SCARA robots are very fast given the flexibility provided by the rotary joints that hold the arm in place
* While they do not take up expansive space to execute their tasks, the arm covers a large area as it works. They can pick and place, over a considerable radius
* They are highly accurate
* Can handle a high payload

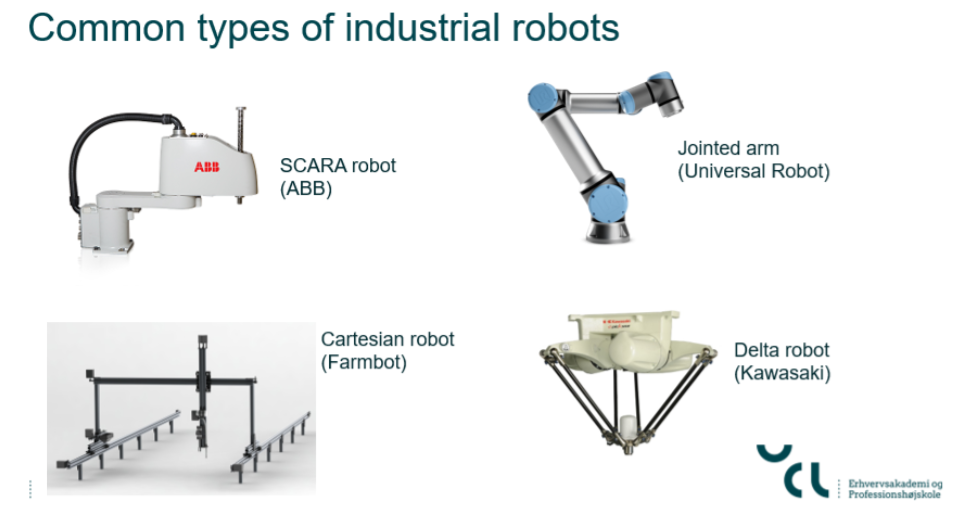
Cons:

* They can only function when placed on or attached to a plane surface thereby offering limited placement flexibility
* Require a dedicated controller

## Industrial Applications

The pick and drop abilities of a SCARA robot make it suitable for:

* Palletizing
* Loading machines
* Assembly functions
* Laser engraving with the aid of an appropriate end-effector
* Soldering



Cobots are also known as collaborative robots as they are designed to co-work with human beings. Traditional robots presented challenges such as collisions with workers in factory spaces. To prevent such accidents, cobots were created as a safer alternative.

# There are three main types of cobots:

## Hand-guiding cobots

They are programmed by technicians using a hand-operated device to train the robot arms on the desired movements.

## Speed and Automation Cobots

They are basically traditional industrial robots with vision systems. They have a warning and a stop zone. When a worker enters the warning zone, they automatically slow down their operation. If they proceed nearer into the stop zone, the cobot stops functioning entirely. They resume operation when both zones are free of obstructions.

## Power and Family Force Limiting Cobots

They are capable of co-working with human beings but are not fitted with any vision systems, scanners, or safety barriers. Instead, they are designed with no exposed motors or sharp corners. In addition, they have collision motors that stop them when there are obstructions around them.

## Safety Monitored Stop Cobots

Compared to other robots they are the least ‘human-friendly’. However, they do have safety sensors that stop and re-engage operation when they detect a human obstruction.



**Standard Gripper 3 Finger Gripper**

Et billede, der indeholder indendørs, køkkenapparat

Automatisk genereret beskrivelseEt billede, der indeholder hjelm

Automatisk genereret beskrivelse

**Magnetic Gripper Soft Gripper**

Et billede, der indeholder indendørs, metal

Automatisk genereret beskrivelse

**Vacuum Gripper Vacuum Gripper with addon**



**Delta Robot (Kawasaki)**

Delta robots were invented by Reymond Clavel and a team of scholars at a college in France. They are also known as parallel robots.

The design of a Delta robot consists of a base mounted overhead that has 3 motors that drive the arms linked to the base. The parallel arms have both prismatic and rotary joints which are not common among other robots. As a result, it can move on the X, Y axes although most users customize their robots to add extra axes. A rotation or pitch axis can both be achieved by modification so as to fulfill certain tasks.

Given as Delta robots are often used for pick and place automation on conveyor systems, programming determines their effectiveness. They are also fitted with cameras which in conjunction with special software make up their vision system. The purpose of the vision system is to differentiate good and bad products during picking and sorting.

Suitable end-effectors are usually attached to the arms, but they require special design schemes. Any additional weight on the arms reduces the high-speed robot arm performance they are favored for. Suction cups are particularly popular with them.

# Pros and Cons

Pros:

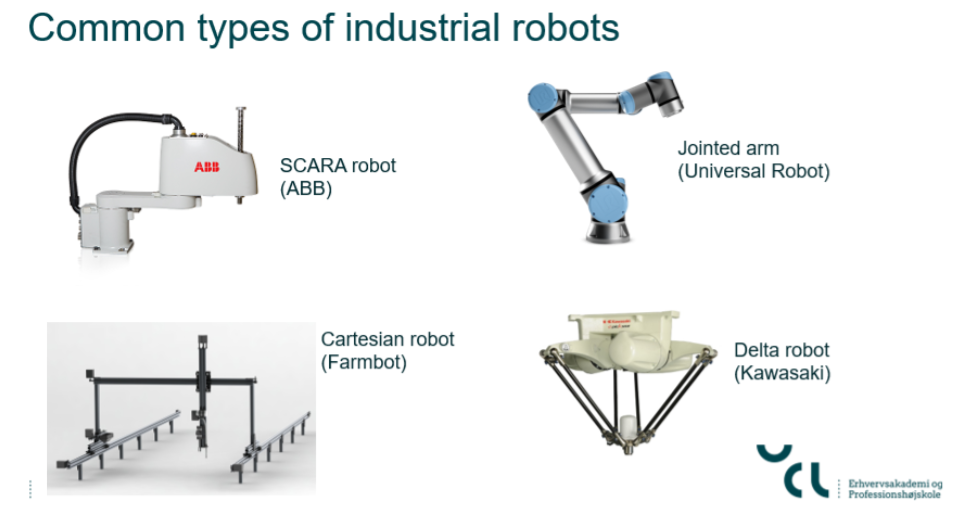
* They are the fastest and even surpass SCARA on the goal-post test
* Can work over a large workspace
* Advanced software makes them very accurate and efficient

Cons:

* Can only manage very light payloads
* High-speed repetitive movements result in wear and tear of parts
* They are expensive

# Industrial Applications

* Flight simulators - this involves an enhanced version that uses 6 additional linear actuators and a movable base
* Pharmaceutical industry
* Sorting product units
* Food packaging
* Low-force assembly tasks



The 3 axes of a Cartesian robot are all perpendicular to each other. Movements are facilitated by:

* 3 prismatic joints that work in sliding motion along the XYZ axes.
* Linear actuators
* Motors to supply power
* A wrist attachment where rotation is needed

They usually come in parts and require assembling by a factory’s machinery experts. While their assembly can be complex it provides a chance for customizations to be done. In fact, it is one of the reasons that make it popular. Precision, speed, and the length of strokes can be adjusted depending on the industry or application.

Cartesian robots are also referred to as Gantry or rectilinear robots. They take up a cubicle shape in terms of their working area. However, one notable characteristic is that they can be mounted in different fashions when necessary. They adapt to overhead mounting as well as vertical and horizontal positions.

# Pros and Cons

Pros

* They can take on a large payload
* Operationally, they are quite versatile
* Affordable
* Can be programmed using online mode
* They have good levels of accuracy

Cons

* Only one axis movement can happen at a time
* They take up a lot of space
* The installation process is generally complex

Industrial Applications

* Handling for plastic molding processes
* Sealing tasks
* 3D printing
* Loading and unloading
* Palletizing tasks

# OPC UA

OPC UA or in other words OPC Unified Architecture is a platform independent service-oriented architecture that integrates all the functionality of the individual OPC Classic specifications into one extensible framework.

This multi-layered approach accomplishes the original design specification goals of:

* **Functional equivalence:** all COM OPC Classic specifications are mapped to UA
* **Platform independence:** from an embedded microcontroller to a could/based infrastructure
* **Secure:** encryption, authentication and auditing
* **Extensible:** ability to add new features without affecting existing applications
* **Comprehensive information modeling:** for defining complex information

## Platform Independence:

Given the wide array of available hardware platform and operating systems, platform independence is essential. OPC UA functions on any of the following:

* **Hardware platforms:** traditional PC hardware, cloud-based servers, PLC, micro-controllers
* **Operating systems:** Microsoft Windows, Apple OSX, Android, any distribution of Linux.

OPC UA provides the necessary infrastructure for interoperability across the enterprise, from machine-to-machine machine-to-enterprise and everything in between.

## Security:

One of the most important considerations in choosing a technology is security. OPC UA is a firewall-friendly while addressing security concerns by providing a suite of controls:

* **Transport:** numerous protocols are defined providing options such as the ultra-fast OPC-binary transport or the more universally compatible JSON over Web sockets.
* **Session Encryption:** messages are transmitted securely at various encryption levels
* **Message Signing:** with message singing the recipient can verify the origin and integrity of received messages
* **Sequenced Packets:** exposure to message replay attacks is eliminated with sequencing
* **Authentication:** each US client and server is identified through X509 certificates providing control over which applications and systems are permitted to connect with each other
* **User Control:** applications can require users to authenticate (login credentials, certificate, web token, etc.) and can further restrict and enhance their capabilities with access right and address-space “views”
* **Auditing:** activities by user and/or system are logged providing an access audit trial

## Extensible:

The multi-layered architecture of OPC US provides a “future proof” framework. Innovative technologies and methodologies such as new transport protocols, security algorithms, encoding standard, or application-services can be incorporated into OPC UA while maintaining backwards compatibility for existing products. UA products built today will work with the products of tomorrow.

**MODBUS:**

Modbus is a data communications protocol originally published by Modicon in 1979 for use with its programmable logic controllers (PLC).

Modbus is popular in industrial environments because it is openly published and royalty-free. It was developed for industrial applications, is relatively easy to deploy and maintain compared to other standards, and places few restrictions – other then the datagram size – on the format of the data to be transmitted.

Modbus supports communication to and from multiple devices connected to the same cable or Ethernet network. For example, there can be a device that measures temperature and another device to measure humidity connected to the same cable, both communicating measurements to the same computer.

Modbus is often used to connect a plant/system supervisory computer with a remote terminal unit in supervisory control and data acquisition systems in the electric power industry.

Communications and devices:

Each device communicating on a Modbus is given a unique address. On Modbus RTU, Modbus ASCII and Modbus Plus, only the node assigned as the Client my initiate a command. All other devices are servers and respond to requests and commands.

Many modems and gateways that support Modbus, as it is a very simple and often copied protocol. Some of them were specifically designed for this protocol. Different implementations use wireline, wireless communication, such as in the ISM band, and SMS or GPRS.

# Documentation

Introduction to PLC: Slides with information and graphical representations of PLC systems <http://www.ieec.uned.es/investigacion/Dipseil/PAC/archivos/introtoplcs_SUPER.pdf>

List of Automation Protocols:

<https://en.wikipedia.org/wiki/List_of_automation_protocols>

UR Robot Simulator and user guide:

<https://www.universal-robots.com/download/software-e-series/simulator-non-linux/offline-simulator-e-series-ur-sim-for-non-linux-594/>

UR Robots inface manual:

<https://s3-eu-west-1.amazonaws.com/ur-support-site/16496/ClientInterfaces_Primary.pdf>

Example of TCP/IP socket connection

<https://forum.universal-robots.com/t/tcp-ip-socket-connection/11378>

Socket programming – Python guide

<https://www.tutorialspoint.com/python_network_programming/python_sockets_programming.htm>

Python Struct library – The module performs conversions between Python values and C structs represented as Python bytes objects

<https://docs.python.org/3/library/struct.html>

How to process package from robot controller – Used for the comment field, where useful posts were found

<https://forum.universal-robots.com/t/how-to-process-package-from-robot-controller/390>

Excell sheet of UR Language:

Et billede, der indeholder bord

Automatisk genereret beskrivelse