What are whycon markers and how do they work?

Whycon markers:
WhyCon is a version of a vision-based localization system that can be used with low-cost web cameras, and achieves millimiter precision with very high performance.

Working:
Even on low-end computers, the core component of the system can detect and estimate 3D positions of hundreds of blackand white markers at the maximum frame-rate of standardcameras.The method is robust to changing lighting conditionsand achieves accuracy in the order of millimeters to centimeters.

What are AruCo markers and how do they work?

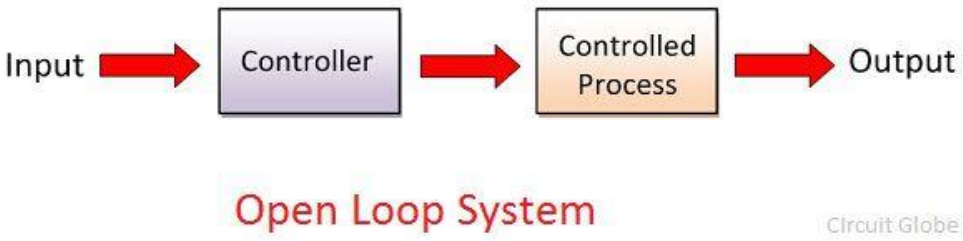
AruCo marker:
AruCo markers are small 2D barcodes. Each AruCo marker corresponds to a number, encoded into a small grid of black and white pixels.

Working:
The AruCo decoding algorithm is capable of locating, decoding, and of estimating the pose of any AruCo markers in the camera's field of view.

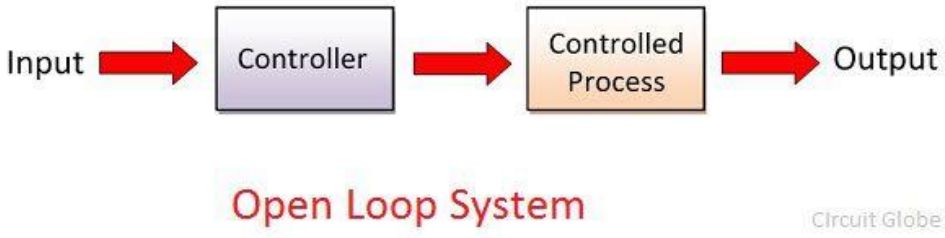
Difference between open-loop and closed loop controllers?

Open loop controllers:
In open loop control system, the output does not affect the control action of the system. In other words, the system whose working depends on time is known as the open loop control system. The open loop system is free from the feedback.

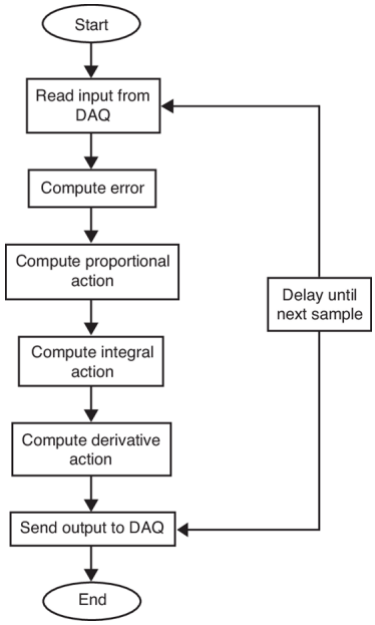
Closed loop controllers:
The closed-loop control system means the output of the system depends on their input. The system has one or more feedback loops between its output and input. The closed-loop system design in such a way that they automatically provide the desired output by comparing it with the actual input.



What are PID controllers?
A proportional–integral–derivative controller is a control loop mechanism employing feedback that is widely used in variety of applications requiring continuously modulated control. A PID controller continuously calculates an error value as the difference between a desired setpoint (SP) and a measured process variable (PV) and applies a correction based on proportional, integral, and derivative terms.



Flow chart to explain the working of a PID algorithm?



Flow chart to explain the working of a PID algorithm?

Optimum way to tune a PID controller?
The best method depend on the desired performance for your application, if reference tracking or disturbace rejection. There are a lot of tuning methods for PID control, the main classic methods being Zigler-Nichols, Cohen-Coon and Chien-Hrones-Reswick. This methods are based on first-order-plus-dead-time models, but have unaltered formulae and don't work for some systems with a small relation (dead-time)/(time constant).

References:

https://core.ac.uk/download/pdf/42583963.pdf
http://jevois.org/moddoc/DemoArUco/modinfo.html
https://circuitglobe.com/difference-between-open-loop-and-closed-loop-system.html
https://en.wikipedia.org/wiki/PID_controller
https://www.researchgate.net/post/What_is_the_best_method_for_tuning_PID_controller_parameters.