INSA INSTITUT NATIONAL DES SCIENCES APPLIQUÉES TOULOUSE

Ball Balancing System

User Manual

ABREU DE ANDRADE Heitor
BELAFKI Ayoub
COELHO ROBL Ana Carolina
EYCHENNE Vincent
GARNICA AZA Jose Eduardo
SUAREZ SEPULVEDA Johan Sebastian

INSA INSTITUT NATIONAL DES SCIENCES APPLIQUÉES TOULOUSE

Table of contents

I. Required MATLAB add-ons

The following free extensions are required:

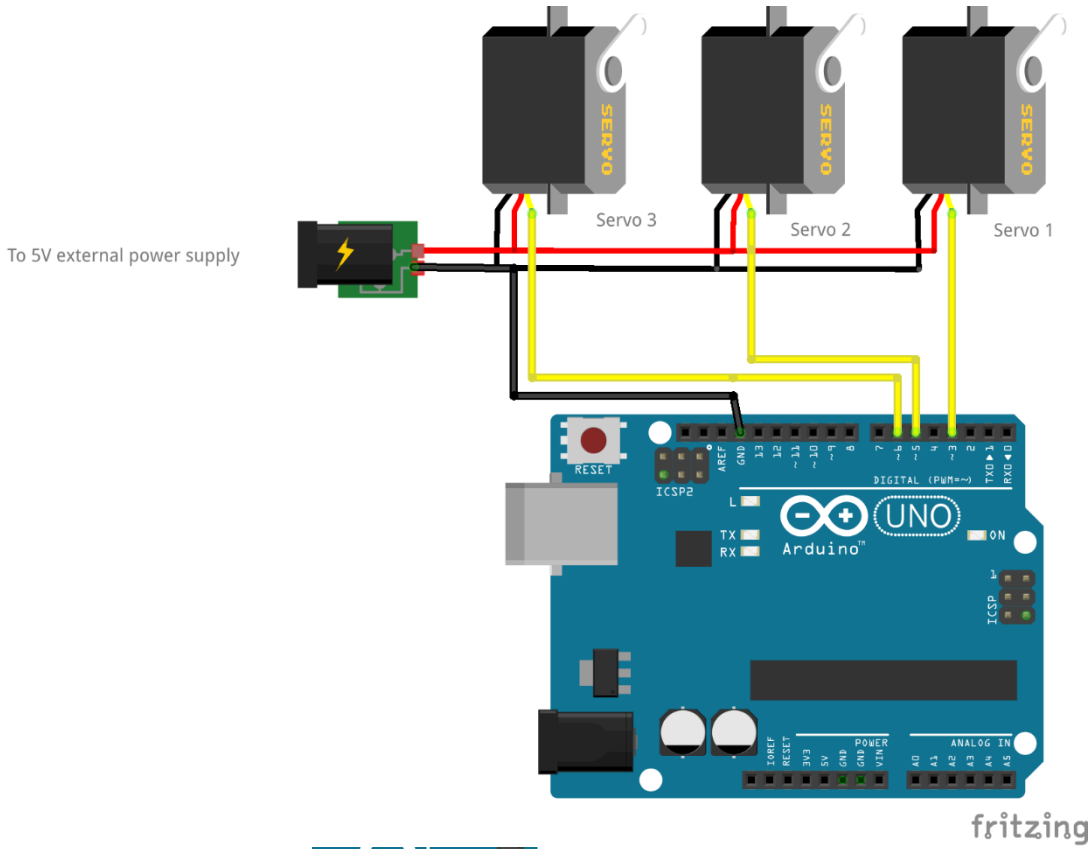
- Simulink Support Package for Arduino Hardware: https://fr.mathworks.com/matlabcentral/fileexchange/40312-simulink-support-package-for-arduino-hardware
- MATLAB Support Package for Arduino Hardware: https://fr.mathworks.com/matlabcentral/fileexchange/47522-matlab-support-package-for-arduino-hardware
- MATLAB Support Package for USB Webcams: https://fr.mathworks.com/matlabcentral/fileexchange/45182-matlab-support-package-for-usb-webcams
- Image Acquisition Toolbox Support Package for OS Generic Video Interface: https://fr.mathworks.com/matlabcentral/fileexchange/45183-image-acquisition-toolbox-support-package-for-os-generic-video-interface?

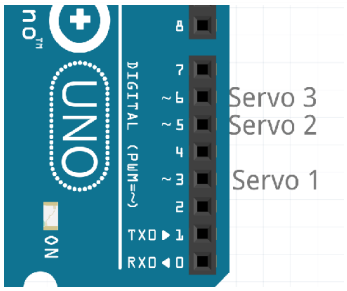
II. Description of the system operation

With the webcam connected to the computer, the position of the ball is determined in real time using the Simulink program **Ball_detection.slx**. The coordinates are then transmitted to the Arduino through the serial port. The Simulink program **PID_Arduino.slx** allows to set the different compensators parameters. The Arduino board is used here as a target by using the "Build, Deploy & Start" function. MATLAB generates C++ code and transfers it to the board.

Important: It is not possible to use the Arduino Uno as an acquisition board. The Uno has only one serial port and this one is already used for the transmission of coordinates!

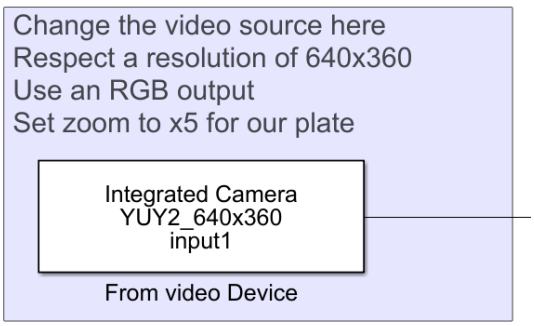
III. System wiring





IV. Setting up Ball_Detection.slx

- Connect the webcam before launching MATLAB!
- Open the “From video device” bloc.

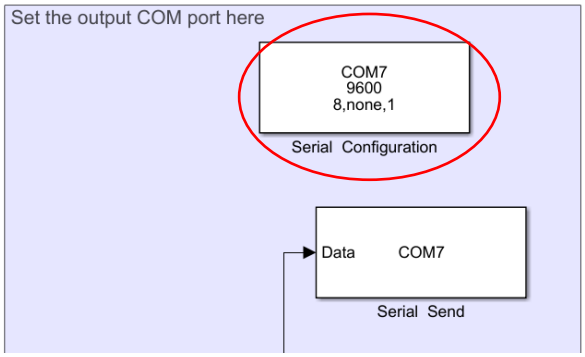


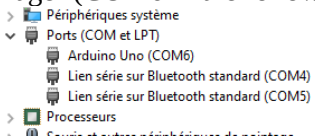
Use the following parameters:

- Device: Choose the external webcam (Microsoft LifeCam HD-3000 here)
- Video format: Set the resolution to 640x360.
- Output color space: RGB.
- Set zoom to x5 by using the “Edit properties” button.

- Open the “Serial Configuration” bloc.

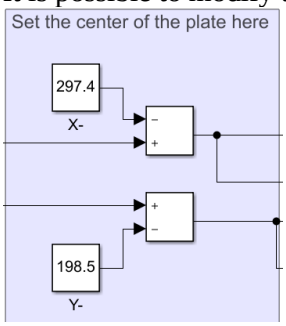
- Select the correct COM port by using the device manager (COM6 in the following example):





- Parameters: Baud rate: 9600; 8 data bits; no parity; 1 stop bit; Byte order: LittleEndian; no flow control
- Open the "Serial Send" bloc and choose the same COM port as in the "Serial Configuration" bloc.

- It is possible to modify the center coordinates here:



- Run the program and do not forget to stop it before transferring new code to the Arduino since it uses the same COM port.