Getting Started with TheButterRobot

Suggest Edits

This page will help you get started with TheButterRobot. You'll be up and running in a jiffy!

The ButterRobot

Introduction

The Butter Robot, further TBR, is an Arduino based project using ROS as an library, e.g. Rosserial-Arduino. This guide is about instullation and exploitation.

Required equipment

- Arduino Mega2560
- OV7670 Camera module
- Resistor set
- 2 DC motors
- Programmer module
- Enegry sourse(battary, etc)

Required Software

Arduino IDE
Intelij Idea
Linux
Ros
Rosserial-Arduino library
Live7670 library

ROS settupping

Skip this step if you have allready installed ROS.
First of all, go to your Linux terminal and type following commands

Text

```
sudo apt-get install software-center
sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu $ (lib_release -sc) main"> /etc/apt/sources.list.d/ros-
sudo apt-get update
```

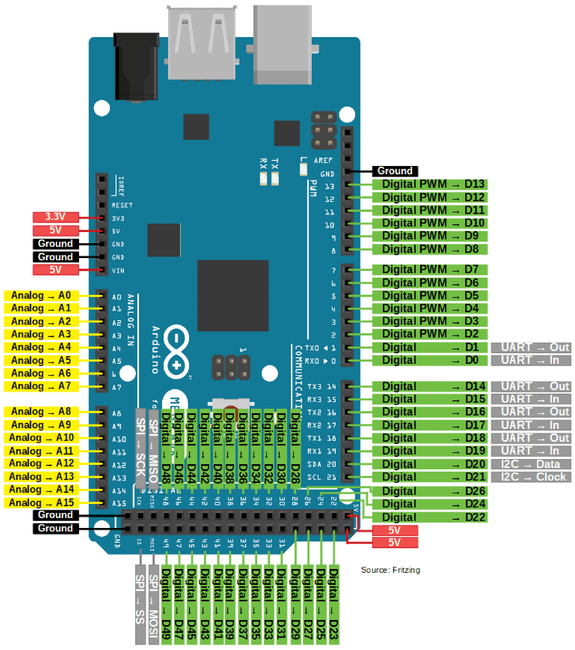
Make sure that you are in the desired directory.

Text

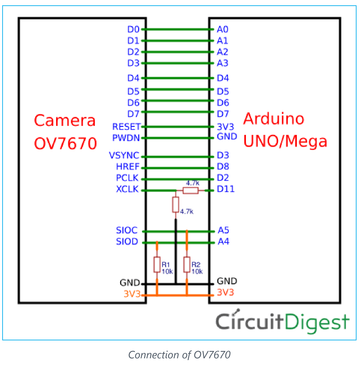
```
sudo apt-get install ros-noetic install
sudo apt install python3-rosdep
sudo rosdep init
source /opt/ros/noetic/setup.bash
```

Pinout and diagrams

Here we present Aduino Mega 2650 pinout. You can use Arduino Uno as well by replacing ports with analogs.



Arduino Mega pinout



Connection of OV7670

Running cameraModule

Install and run LiveOV7670 by calling \LiveOV7670-master\src\LiveOV7670\LiveOV7670.ino file. In the file ExampleUart.cpp you can choose UART_MODE depending on Arduino module and wishing parametrs. In this project we will mostly use UART_MODE 2 or UART_MODE 6 for monochrome images.
Then install and run project ArduinoImageCapture. By choosing your port and baud of UART_MODE you will get images. The first frame is colored in green or red. Green means that all connections set successfully.
To save images it is required to choose a folder.

Recognizing objects in images

Start any python machine and upload model file. Upload your image and run code below. As a result you should get an image and list of objects detected on it.

Python

```
cv2.setUseOptimized(True);
ss = cv2.ximgproc.segmentation.createSelectiveSearchSegmentation()
results_list = []
orig_img = cv2.imread("path"+filename)
ss.setBaseImage(orig_img)
ss.switchToSelectiveSearchFast()
ssresults = ss.process()
imout = orig_img.copy()
```