



OV7670 Arduino Camera Sensor Module Framecapture Tutorial

by **mybotic** on October 20, 2016

Table of Contents

OV7670 Arduino Camera Sensor Module Framecapture Tutorial	1
Intro: OV7670 Arduino Camera Sensor Module Framecapture Tutorial	2
Step 1: Material Preparation	2
Step 2: Pinout Connection	3
Step 3: Java	3
File Downloads	4
Step 4: Sample Source Code	4
File Downloads	5
Step 5: Checking your port	5
Step 6: Com_#.rar	6
File Downloads	6
Step 7: C:\out	7
Step 8: Extra.rar	7
File Downloads	8
Step 9: CMD	8
Step 10: Result	9
Step 11: Troubleshooting	10
Related Instructables	11
Advertisements	11
Comments	11

Intro: OV7670 Arduino Camera Sensor Module Framecapture Tutorial

Description

The camera module is powered from a single +3.3V power supply. An external oscillator provide the clock source for camera module XCLK pin. With proper configuration to the camera internal registers via I2C bus, then the camera supply pixel clock (PCLK) and camera data (Data[9:0]) back to the host with synchronize signal like HREF and VSYNC. The OV7670 camera module is a low cost 0.3 mega pixel CMOS color camera module, it can output 640x480 VGA resolution image at 30fps. The OV7670 camera module build in onboard LDO regulator only single 3.3V power needed and can be used in Arduino, STM32, Chipkit, ARM, DSP , FPGA and etc.

Specification

- Optical size 1/6 inch
- Resolution 640x480 VGA
- Onboard regulator, only single 3.3V supply needed
- Mounted with high quality F1.8 / 6mm lens
- High sensitivity for low-light operation
- VarioPixel® method for sub-sampling
- Automatic image control functions including: Automatic
- Exposure Control (AEC), Automatic Gain Control (AGC), Automatic White Balance (AWB), Automatic
- Band Filter (ABF), and Automatic Black-Level Calibration (ABLC)
- Image quality controls including color saturation, hue, gamma, sharpness (edge enhancement), and anti-blooming
- ISP includes noise reduction and defect correction
- Supports LED and flash strobe mode
- Supports scaling
- Lens shading correction
- Flicker (50/60 Hz) auto detection
- Saturation level auto adjust (UV adjust)
- Edge enhancement level auto adjust
- De-noise level auto adjust

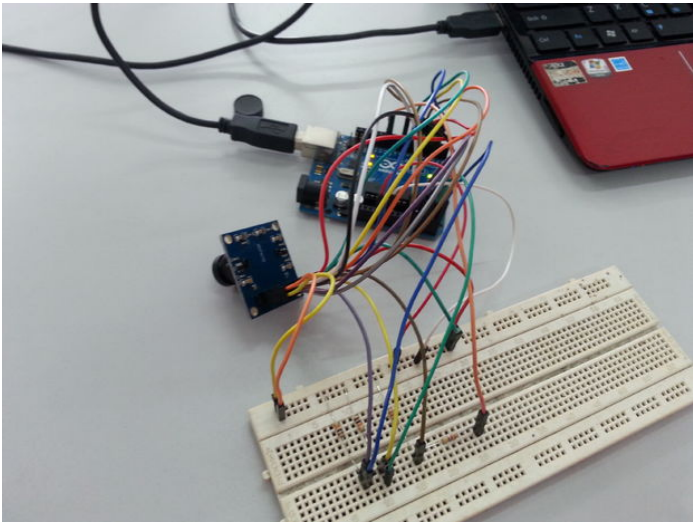


Step 1: Material Preparation

In this tutorial, you will need :

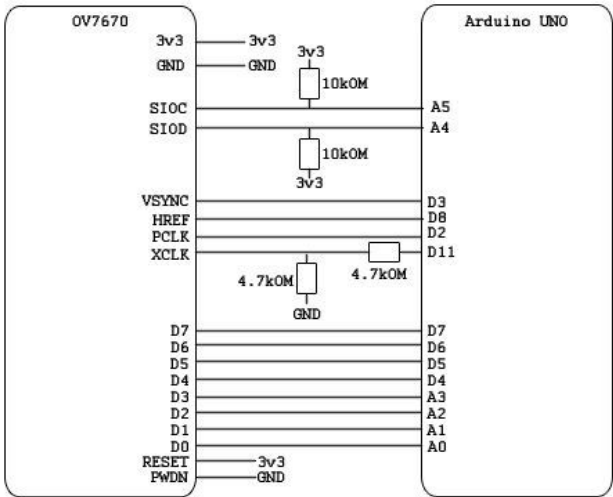
1. Arduino Uno Board and USB
2. OV7670 Arduino Camera Sensor Module STM32
3. 10K resistor X 2
4. 4.7K Resistor X2
5. Breadboard

and you will be using Arduino IDE and Java.



Step 2: Pinout Connection

Connect your circuit as illustrated on the picture above.

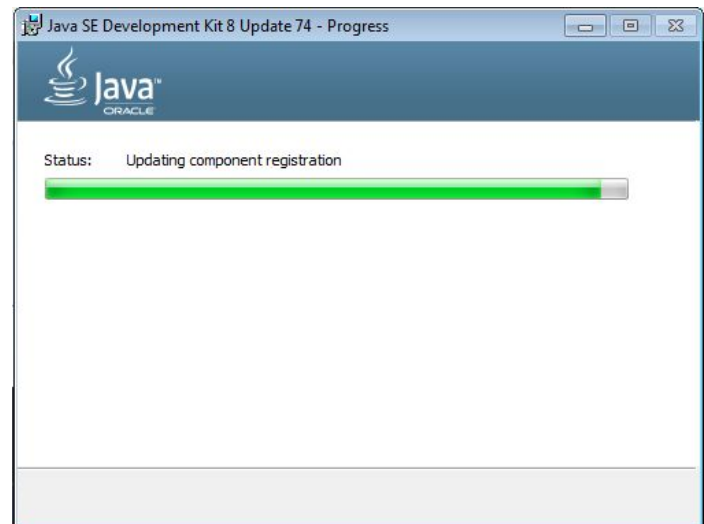
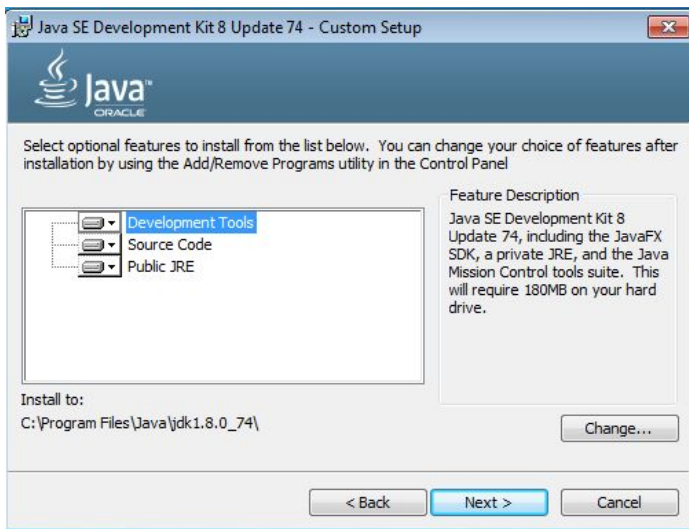
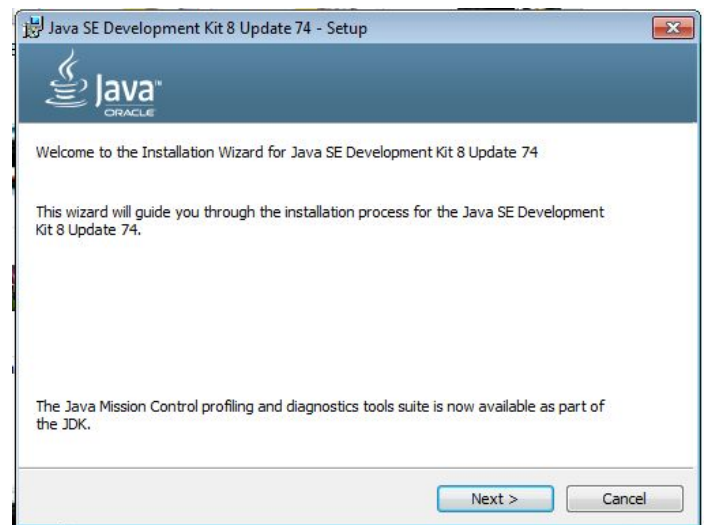
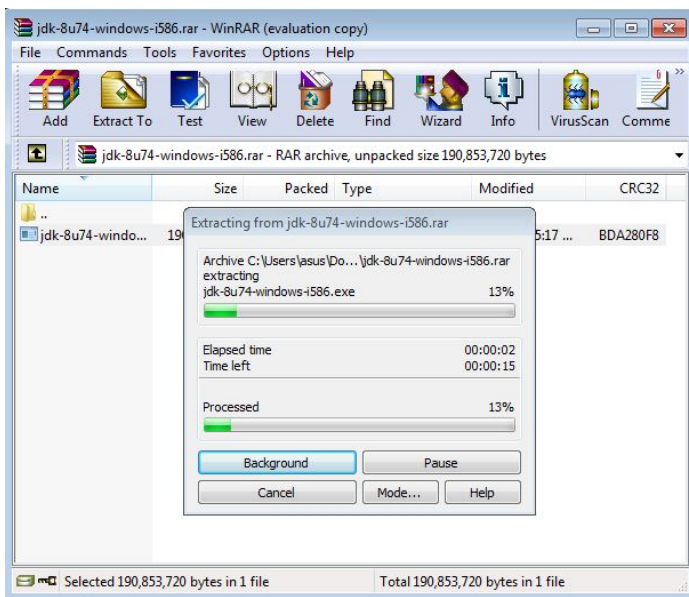


Pin	Type	Description
VDD**	Supply	Power supply
GND	Supply	Ground level
SDIOC	Input	SCCB clock
SDIOD	Input/Output	SCCB data
VSYNC	Output	Vertical synchronization
HREF	Output	Horizontal synchronization
PCLK	Output	Pixel clock
XCLK	Input	System clock
D0-D7	Output	Video parallel output
RESET	Input	Reset (Active low)
PWDN	Input	Power down (Active high)

Step 3: Java

You have to download this to be able to interface this camera sensor module on your pc. Open the zip file and install to your computer. By default, this will be install to your c drive in your program files folder. As for mine, it is stored in C:\Program Files\Java\jdk1.8.0_74. You can refer picture above.

NOTE : This jdk-8u74-windows-i586.rar file is quite big.



File Downloads



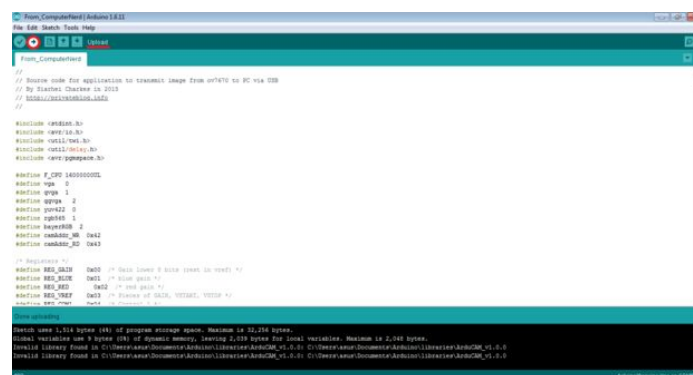
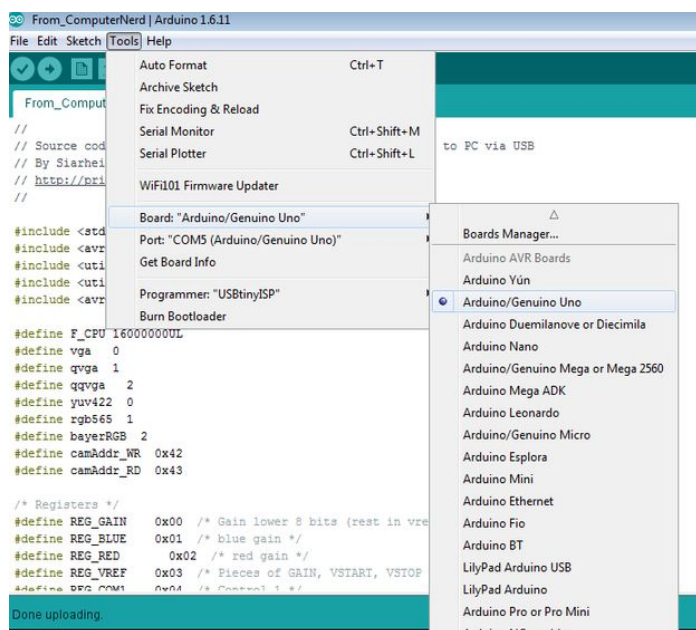
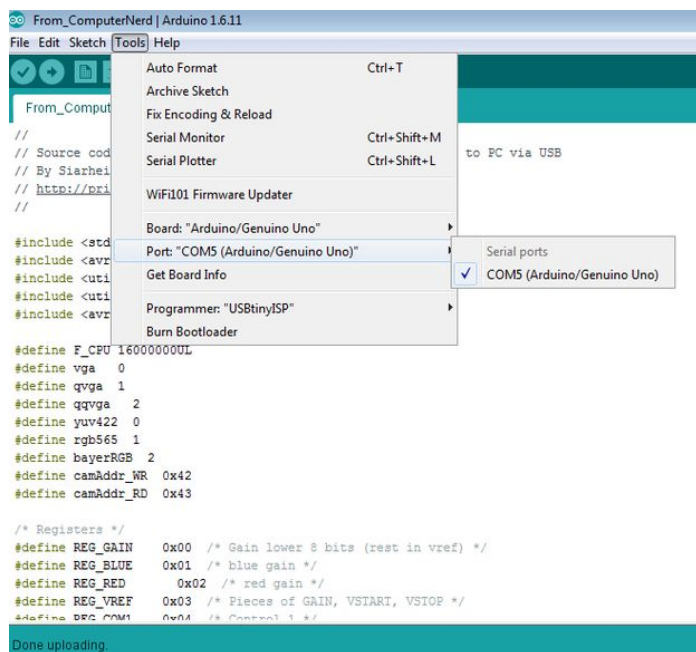
jdk-8u74-windows-i586.rar (180 MB)

[NOTE: When saving, if you see .tmp as the file ext, rename it to 'jdk-8u74-windows-i586.rar']

Step 4: Sample Source Code

This sample source code is originally from this link. Download this sample source code, open it on your Arduino IDE, connect your Arduino Uno Board to your computer and upload it. Select the correct board and port before uploading this source code. Refer image above for help.

NOTE : Don't mind that it does not recognize some libraries in the beginning of the file. If it bothers you, comment the whole line out with //



File Downloads

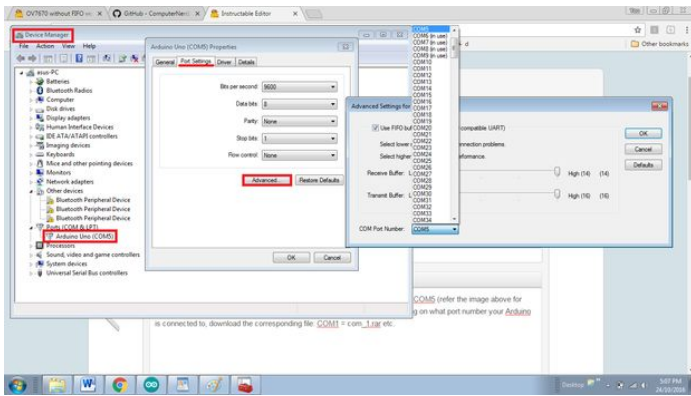


From_ComputerNerd.rar (5 KB)

[NOTE: When saving, if you see .tmp as the file ext, rename it to 'From_ComputerNerd.rar']

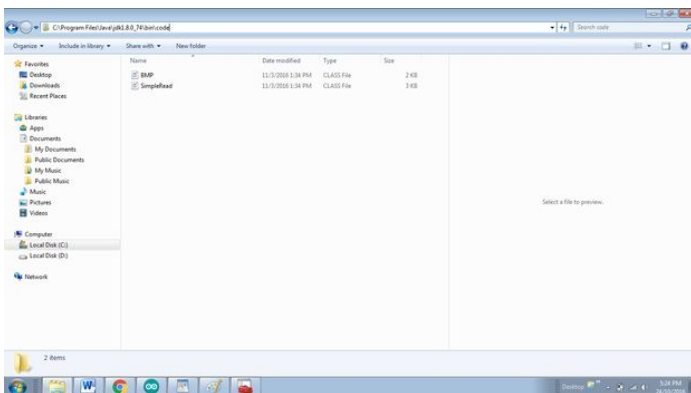
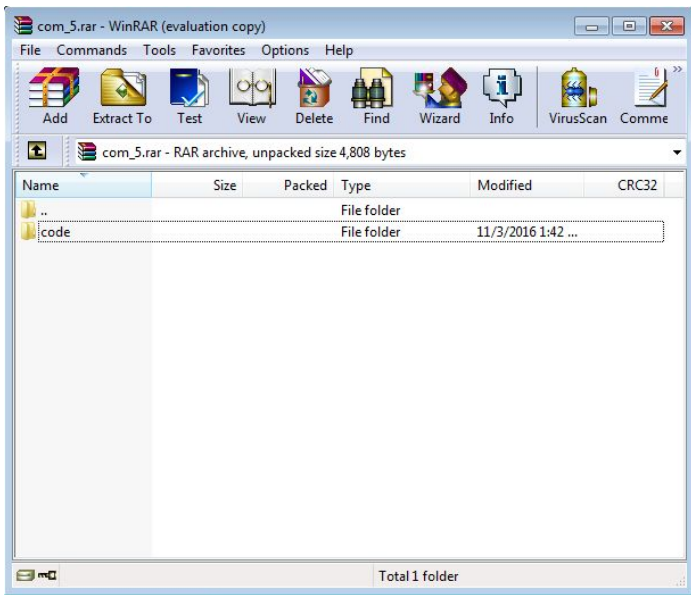
Step 5: Checking your port

Find out what USB port your Arduino is connected to, something like COM4 or COM5. You could check it on your Control Panel > Device Manager > Ports (COM & LPT) > right click > Properties > Port Settings > Advanced > COM Port Number > /Select Your Port Number/. You can refer the image above for help. As for mine, it is port COM5.



Step 6: Com_#.rar

Depending on what port number your Arduino is connected to, download the corresponding file. COM1 = com_1.rar , COM2 = com_2.rar and etc. Open your com#.rar on WinRAR, extract/copy the 'code' folder from com_#.rar, go to Local Disk > Program Files > Java > jdk1.8.0_74 > bin and paste it there. you can refer the picture above for a better understanding.



File Downloads



com_1.rar (2 KB)

[NOTE: When saving, if you see .tmp as the file ext, rename it to 'com_1.rar']



com_2.rar (2 KB)

[NOTE: When saving, if you see .tmp as the file ext, rename it to 'com_2.rar']



com_4.rar (2 KB)

[NOTE: When saving, if you see .tmp as the file ext, rename it to 'com_4.rar']



com_3.rar (2 KB)

[NOTE: When saving, if you see .tmp as the file ext, rename it to 'com_3.rar']

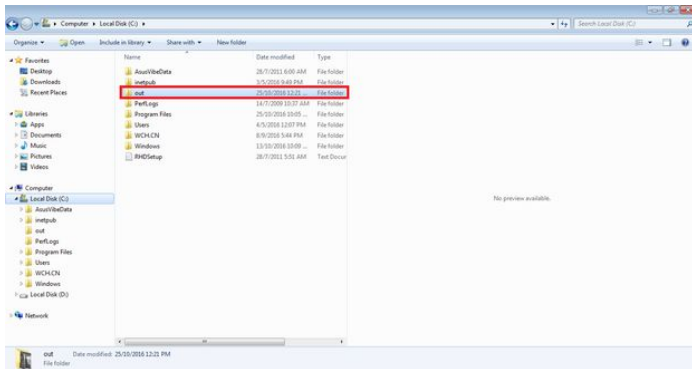


com_5.rar (2 KB)

[NOTE: When saving, if you see .tmp as the file ext, rename it to 'com_5.rar']

Step 7: C:\out

You have to create a new folder on your c drive / Local Disk (C:). This will be your camera module sensor's taken picture's destination. Name this folder as out (it could be anything though).

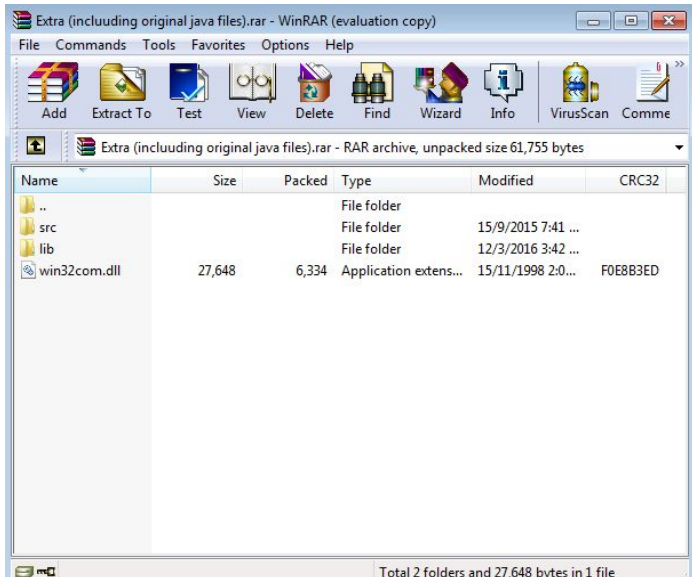


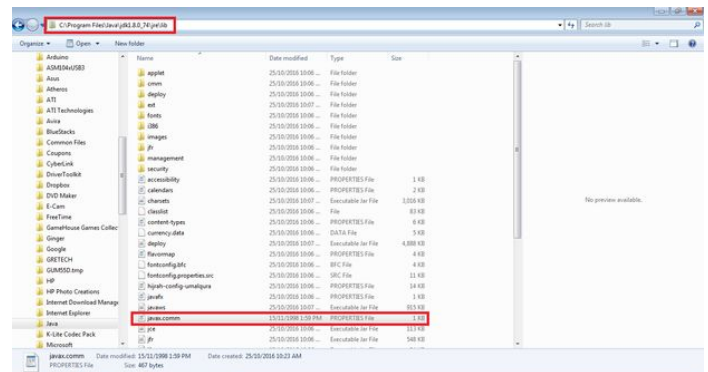
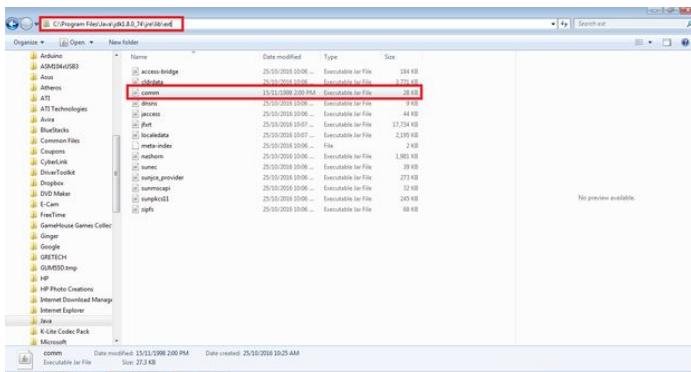
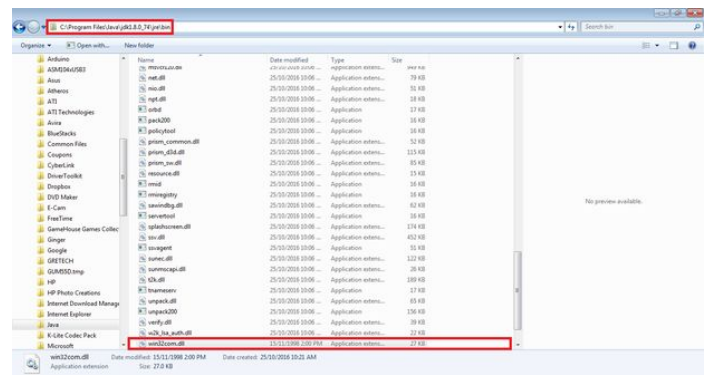
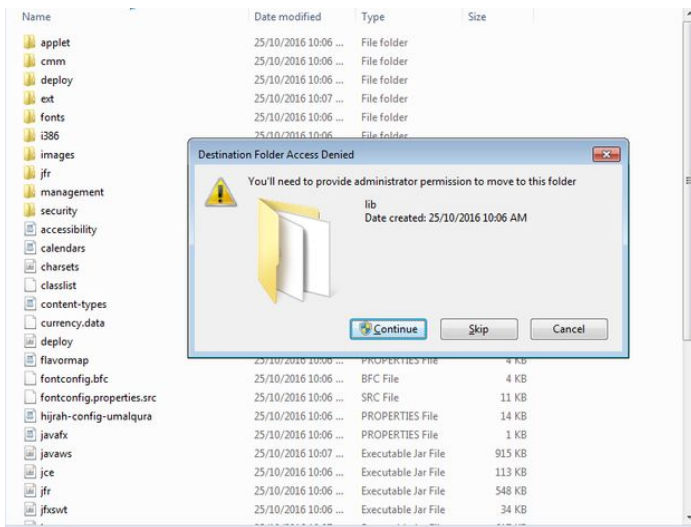
Step 8: Extra.rar

1. Download and extract the Extra.rar file below and open it. As you could see, there is src folder, lib folder and "win32com.dll". You have to copy and placed "win32com.dll" in the "C:\Program Files\Java\jdk1.8.0_74\jre\bin" directory.

2. Then, open the lib and you could see the "comm.jar" and "javax.comm.properties" in it. Copy and paste the "comm.jar" in "C:\Program Files\Java\jdk1.8.0_74\jre\lib\ext" and "javax.comm.properties" in the "C:\Program Files\Java\jdk1.8.0_74\jre\lib" directory.

NOTE : it may ask for administrator permission, just proceed anyway.





File Downloads



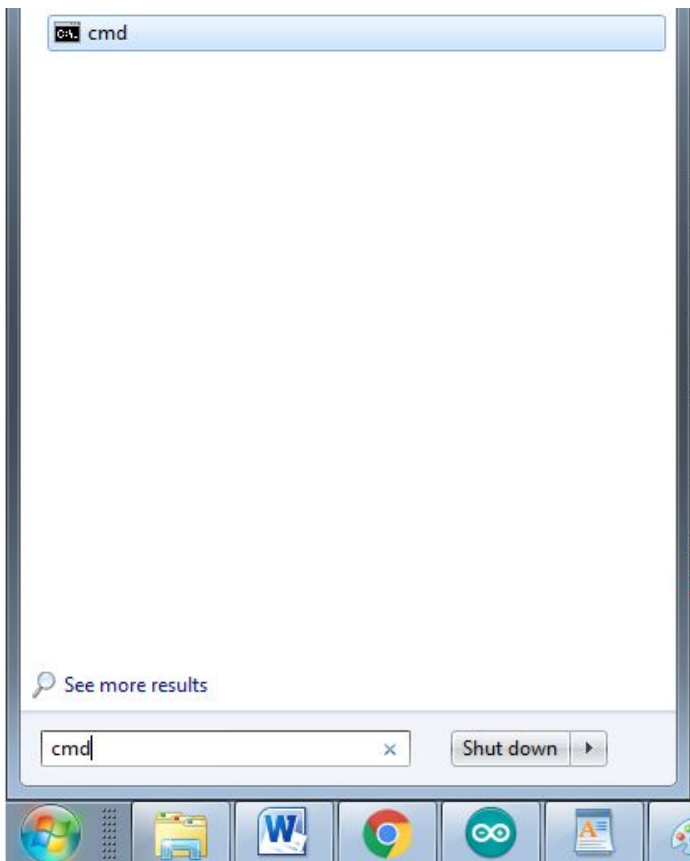
Extra (including original java files).rar (33 KB)

[NOTE: When saving, if you see .tmp as the file ext, rename it to 'Extra (including original java files).rar']

Step 9: CMD

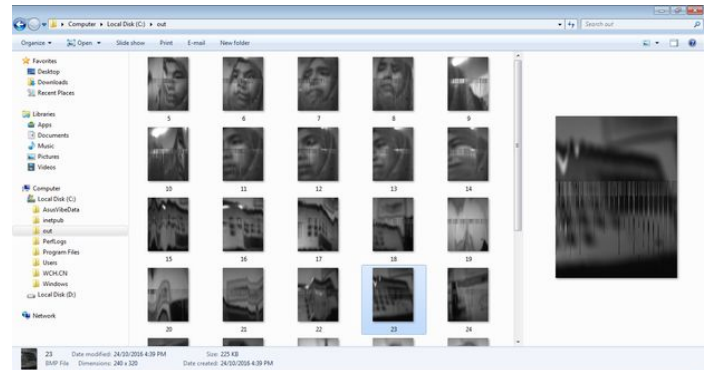
Open cmd and open the folder wherein the code folder is by writing its address. mine is "C:\Program Files\Java\jdk1.8.0_74\bin". Click enter and it will give u "C:\Program Files\Java\jdk1.8.0_74\bin>". Now, write "java code.SimpleRead". It supposed to be like this----> C:\Program Files\Java\jdk1.8.0_74\bin>java code.SimpleRead. You can refer image for help.

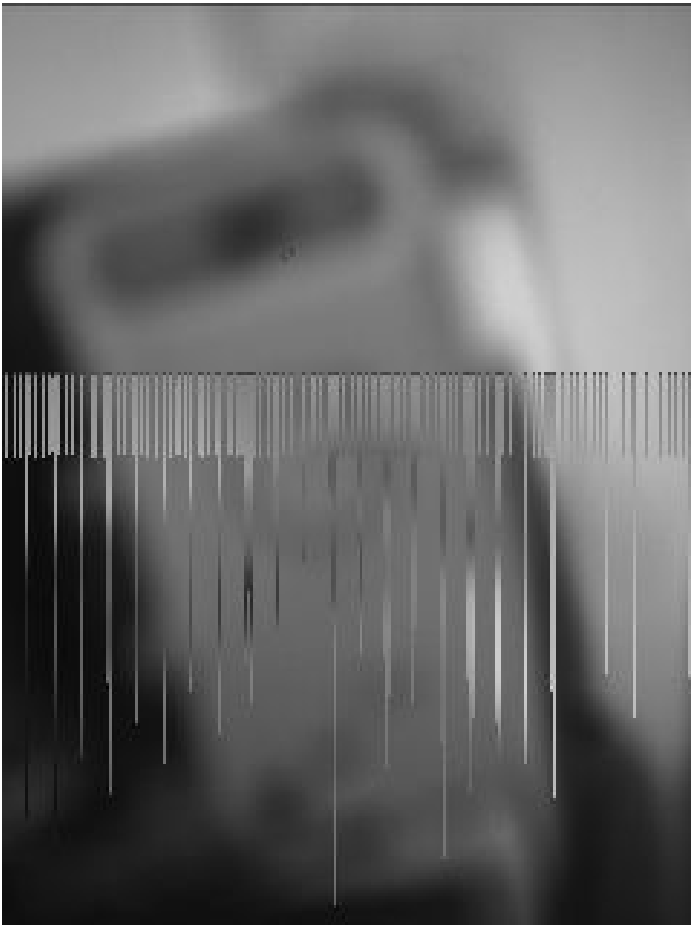




Step 10: Result

You can see the picture taken from your OV7670 Camera Module Sensor in your our folder in c drive.

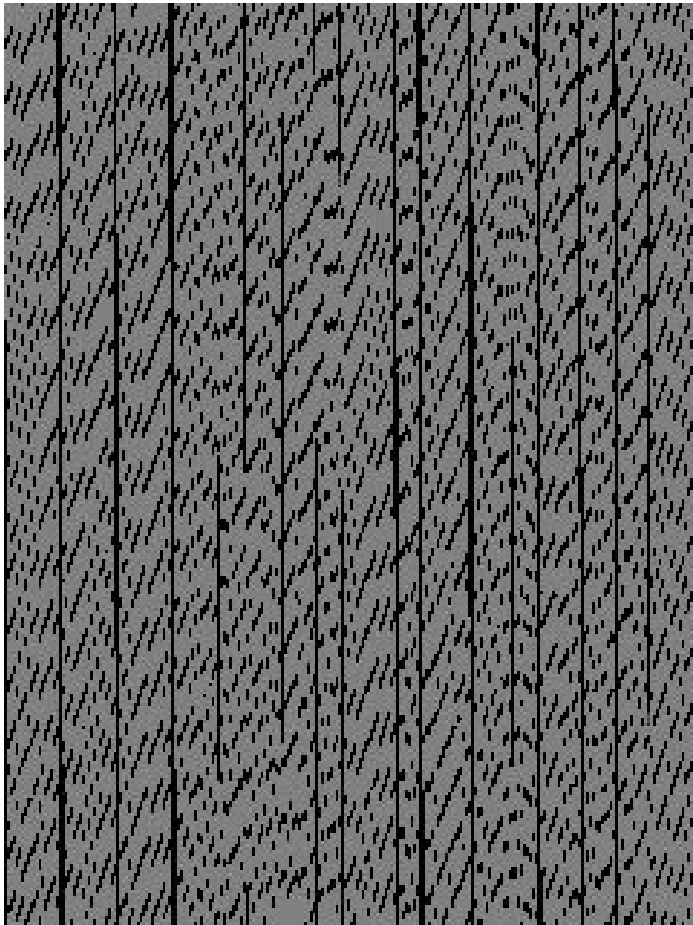




Step 11: Troubleshooting

1. If your cmd cannot find images, try switching the USB of the arduino from a USB2.0 to a USB3.0 (blue socket) or vice versa.

2 If you receive scrambles pictures, you need to edit the sample source code. Edit line: `wrReg(0x11, 12)`; try changing the 12 to 11/10/9/13. Upload it to the arduino and run the cmd code again.



Related Instructables



How to use OV7670 Camera Module with Arduino? by ElecFreaks



OV7670 without FIFO very simple framecapture with Arduino, black/white by Kepp27



How to connect OV7670 to Arduino Due by Dennisv15



FPGA Robotics Vision for anyone by Dennisv15



Connect Camera to Zedboard by srs_sput



DIY STC MCU Combined Wireless Small Camera with OV7670, NRF24L01 Wireless Transceiver Module, 2.8" TFT LCD (Provide code) by ICStation

Comments