

# ELECTRONIC CORRECTOR

Y.Deepak Kumar<sup>1</sup>, D.Pradeep<sup>2</sup>

<sup>1,2</sup>Sri Venkateswara College of Engineering, Affiliated to Anna University, Post Bag No. 3, Pennalur, Sriperumbudur. Pincode-602105.

---

**Abstract:-** It is natural that everyone makes mistakes while writing on paper. Natural choice is a whitener to make a correction. But not all paper color matches with that of the whitener ink. So sometimes, it might make our work look shabby. This project aims towards creating a whitener, or rather a corrector, that generates the ink which matches with the color of the paper. The primary color mixture is identified for the captured background color. This color is created by mixing primary colors, y, cyan, magenta and yellow, by controlling solenoid valves, which are controlled by arduino controller. This ink is applied on the desired medium to make the required correction. All combinations of colors are obtained by suitably mixing the CMY primary colors in appropriate proportions. This is a novel idea that was created on personal interest.

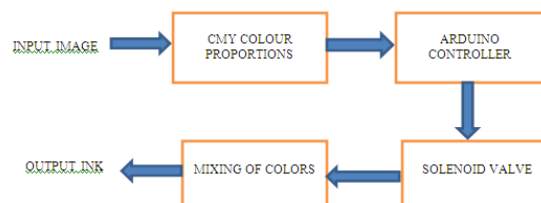
**Keywords:-** MATLAB, ARDUINO, CMY, ink color, solenoid valves.

---

## I. INTRODUCTION

People usually are prone to making mistakes when writing on paper. To cover it up, a whitener is used. But a whitener's ink color does not always match with the paper color. Thus, the correction looks distinct on a piece of paper which is not so much appealing. Thereby, this article aims to unleash our novel idea to create a device that will produce ink according to the paper color. The image is captured, processed and the desired color is produced using CMY colors.

## II. STEPWISE PROCEDURE



**Fig 1.** Block Diagram

### A. Image acquisition:

The image is acquired by a camera, which is used to capture the paper color alone without considering the written text. The camera is actuated using MATLAB image processing tools.

### B. Primary color proportions:

The RGB (Red, Green and Yellow) color proportions are found for the captured image using MATLAB. RGB are the primary colors of light. Thereby, mixing these colors in pigments will not give the required color. The primary colors of pigments are CMY (Cyan, Magenta and Yellow). Hence RGB is converted to CMY and these values in proportions are fed to the controller.

### C. Arduino controller and solenoid valves:

Arduino controller is interfaced with MATLAB to control the flow of colors from the solenoid valves. Any other controller may be used for this application, but Arduino is preferred for its simplicity. Based on the proportions of CMY from MATLAB, the solenoid valves are actuated by the Arduino accordingly to control predefined buffers containing CMY colors separately. The CMY colors are channelled separately in a mixer unit.

### D. Color mixing:

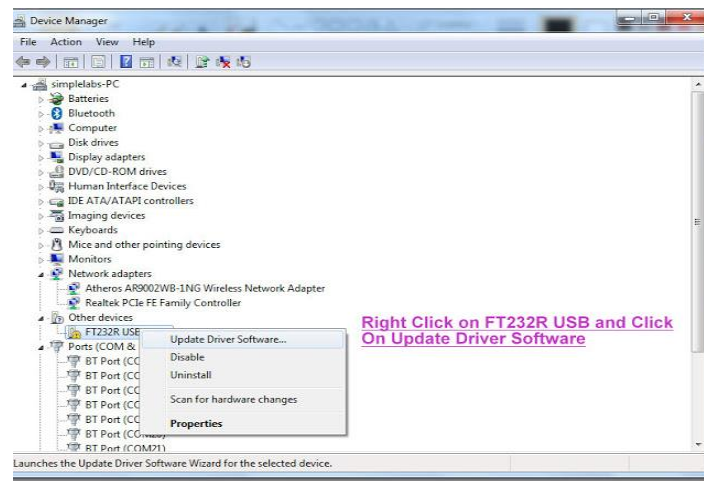
The colors which are obtained from the buffers based on the proportions are thoroughly mixed in a mixer unit using a motor. After mixing, the desired color is obtained. This color matches with that of the paper color and is therefore ready to be applied on the paper.

### III. INITIAL CONFIGURATION OF ARDUINO

Arduino is not a plug and play device. It needs to be configured before we can use it for the first time, so that the PC can recognise the presence of an external device in one of its ports. Also the port number to which the arduino is connected needs to be found out.

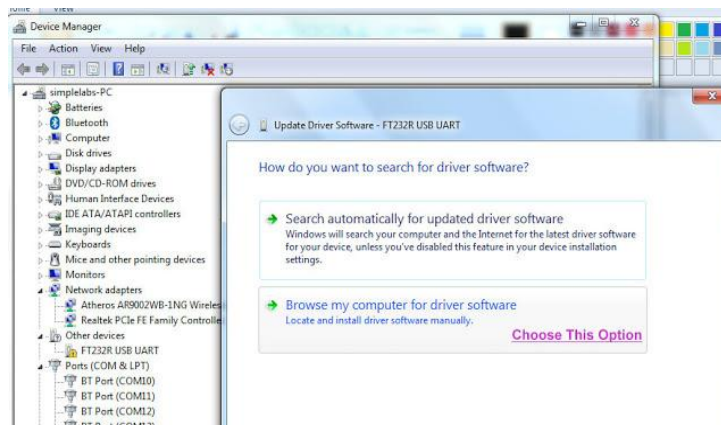
A. Steps for initial configuration:

1. The latest version of Arduino software needs to be downloaded from the official arduino website, which is =><http://arduino.cc/en/Main/Software>
2. The files are in zipped format, which are unzipped and the location is to be remembered, as it will be used in the forthcoming steps.
3. Arduino board is connected to PC using the USB cable provided. The power (red) led will go 'ON' the board.
4. Windows will detect a new USB device. Automatic installation of any drivers must be ignored.
5. The device manager is opened by right clicking on 'My Computer' and choosing Properties. In the new Window, Device Manager is selected.
6. 'FT232R USB UART ' will have a warning icon next to it as the drives have not been installed. The 'Update Driver Software...' option is selected by right clicking on it.



**Fig 2.**

7. Now, Windows will provide two options. "Locate and install drivers manually from a specific Location." is chosen.



**Fig 3.**

8. In the next screen, the locations of the drivers are selected and then 'OK' is selected. Then 'Next' is selected.

10. Now, Windows might throw a Security warning, which is ignored and next is selected.

11. The Device will be installed as a USB Serial Converter

12. Now Windows will detect a USB Serial Port and ask for drivers. The exact same process as did earlier needs to be followed, choosing the same driver location as earlier.

In case Windows doesn't ask you, scroll through the Device manager and find the listing called 'USB Serial Port' and follow the above steps.

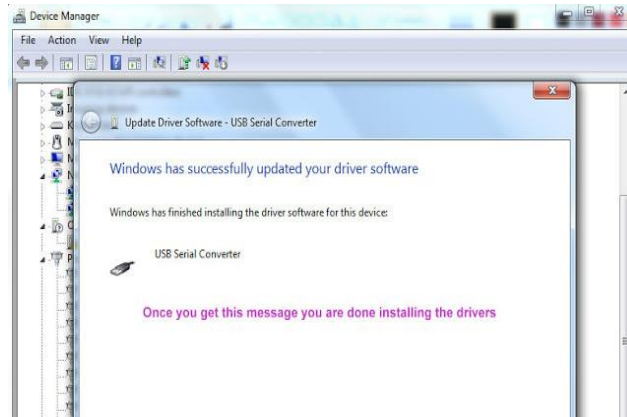


Fig 4.

13. Once everything is finished, a new device 'USB Serial Port (COMXX)' can be seen. This is the port number which needs to be noted down.



Fig 5.

14. After these steps, the arduino is ready to be programmed, as it has been configured.

#### IV. MATLAB INTERFACING WITH ARDUINO

Now MATLAB needs to recognize the ARDUINO. For this, the arduino IO files are added into the MATLAB library. Once this is done, the ARDUINO can be directly controlled by MATLAB. Initially, an object needs to be created to denote the port number, and then this port must be opened to enable communication. Next, a particular coordinate of the input image is selected and the RGB values are found out. For example,

Red= image (100, 200, 1)

Green= image (100, 200, 2)

Blue= image (100, 200, 3)

where "image" is a variable which contains the image.

As mentioned before, since the required values are those of CMY, the RGB values are converted into CMY values using the formulae,

Cyan= 255-red

Magenta=255-green

Yellow=255-blue

These CMY values are normalized for easier calculations. Once these values are obtained, we need to control the ARDUINO. Using the object, which was already created, direct commands can be given to the ARDUINO, so that it can give the output voltage for a required amount of time, which depends on the CMY values.



Fig 6. Example for MATLAB output.

Brown color was given as the input image and the RGB values and the normalised CMY values were obtained. An Arduino board consists of an 8-bit Atmel AVR microcontroller with components to facilitate programming and incorporation into other circuits. Official Arduino have used the mega AVR series of chips, specifically the ATmega8, ATmega168, ATmega328, ATmega1280, and ATmega2560. ATmega 328 is a 28 pin IC. 20 pins can be used for providing the output, out of which 14 can provide digital I/O pins, and 6 pins provide analogue inputs. Most boards include a 5 volt regulator and a 16 MHz crystal oscillator.

## V. SOLENOID VALVE INTERFACE

The values from MATLAB fed to ARDUINO are used to control solenoid valves. The solenoid valves require 12v DC supply, whereas the ARDUINO output is 5v. So, relays are used to drive the valves. The solenoid valves are attached to cyan, yellow and magenta color ink filled in their buffers. Based on the values obtained from MATLAB, respective solenoid valves are actuated for a particular time period, letting the colored ink flow for the determined color values. These color inks are obtained from solenoid valves and are fed to a mixing chamber. The chamber contains a motor which starts after the flow process is completed. The motor is allowed to run for a predefined duration controlled by ARDUINO. After the mixing process, the ink is ready to use and is found to match the color of the paper.

## VI. CONCLUSION

This product is developed to be used on any colored paper to camouflage the correction and color of the paper. This product could be extensively used by students in their craft work; also it can be a solution to different colored paints where the same could be generated by giving the input as the desired color in the product. This application could be also used on important government documents where replacement may be costly. By using printer technology, application specific programs and integrated chips, miniaturization could be achieved. Automation could further be developed by detecting the presence of paper and automatic capture of the image by using pressure sensor in the device tip, detecting the pressure and capturing it while placing it on the paper. In order to improve the quality of the image high resolution camera must be used which may prove costly and the light intensity on the paper will also play a major role in deviation from the actual color. Though the world is moving to a digital era, this device will widely be accepted in the near future by all.

## REFERENCES

- [1] Hashim, Shah Alam Rahman, R.A. Jarmin, R.” A Study on RGB Color Extraction of Psoriasis Lesion using Principle Component Analysis” IEEE TRANS. Research and Development, 27-28 June 2006.
- [2] Liang Chu; Yanli Hou; Minghui Liu; Jun Li; Yimin Gao; Ehsani, M. “.Study on the Dynamic Characteristics of Pneumatic ABS Solenoid Valve for Commercial Vehicle” Vehicle Power and Propulsion Conference, 2007. IEEE Topic(s): Components, Circuits, Devices & Systems ; Transportation
- [3] Ying-bao Zhao. “Solenoid Valve Respond Time Test System Basing on Single Chip Micro-computer”. IHMSC '09. 26-27 Aug. 2009.
- [4] Eddins, S.L.; Orchard, M.T. “Using MATLAB and C in an image processing lab course”. Computing & Processing (Hardware/Software); Signal Processing & Analysis.. ICIP-94., IEEE International Conference Volume: 1, 1994, Page(s): 515 - 519.
- [5] Gossett, N.; Chen, B. “Paint Inspired Color Mixing and Compositing for Visualization “.Information Visualization, 2004. INFOVIS 2004. IEEE Symposium on Topic(s): Computing & Processing (Hardware/Software); Signal Processing & Analysis. 2004 , Page(s): 113 - 118
- [6] <http://www.mathworks.in/academia/arduino-software/arduino-matlab.html>
- [7] <http://www.solenoid-valve-info.com/solenoid-valve-coils.html>
- [8] <http://www.ladyada.net/learn/arduino/lesson1.html>
- [9] <http://playground.arduino.cc/Learning/SolenoidTutorial>