# *Digital Color Sensor based Solution for Recyclable material Sorting System*

Florin Drăgan, Rodica Holonec, Romul Copindean

Faculty of Electric Engineering, Technical University of Cluj-Napoca, Romania

***Abstract:*** *This paper describes a microcontroller based application allowing color analysis of several materials and remote monitoring of a sorting system. The solution presented provides data acquisition for color temperature and object proximity and transmits those parameters to a main system, displaying of these parameters. Application also allows locally control and data exchange using communication protocols, such as ASCII serial streaming or, TCP/IP over Ethernet. The solution consists of hardware and software components. Hardware component consist of an ARM microcontroller (STM32 family), as CPU main unit, digital color sensor, communication interface and a block of several MOSFET based switches, that are used to control solenoids of pressured air valves. Pieces of plastic materials that need to be sorted sequentially pass through a conveyor belt and are analyzed by the color sensor. Depending on color, each piece of material will be blowed off to appropriate section of a clustered container. After each puff, controller transmits a data frame to the main control unit, providing several information about color and number of the analyzed pieces. Another optical sensor detects the presence of pieces at first of the bend, allowing controller to transmit the start and stop commands to the inverter that control the conveyor's motor. Communication is provided by using RS485 serial interface. This application example works as stand alone but can be part of a complex Recyclable Materials Management and Sorting System.*

***Keywords:*** *Recyclable Materials Management and Sorting System, microcontroller, digital color sensor, inverter control using communication protocol*

## 1. INTRODUCTION

As is well known, one of the WEEE *(Waste Electrical and Electronic Equipment)* Directive (2012/19 / EU) refers to the appropriate recycling of electrical and electronic equipment waste, as valuable materials. The objectives of this directive are to preserve, protect and improve the quality of the environment, protect human health and use natural resources in a prudent and rational manner.

In PCB recycling systems two principles are used: the selective sorting based on "inspection and sorting" stages and simultaneous sorting based on "evacuation and sorting" stages [1] [2].

This paper describes a digital optical sensor based solution allowing medium size component sorting. Electronic components, like coils, transformers, connectors, buttons and other similar medium size plastic made pieces features (but not small resistors or capacitors, SMD type) can be identified by colors, or by magnetic. The magnetic features based sorting solution is not an objective of this paper.

The presented sorting system comprises a conveyor belt, actuated by a three-phases variable frequency inverter driven motor and an ARM microcontroller based processing and control unit.

The control and processing unit uses, for detecting and analyzing objects to be sorted, one digital color and one optical proximity sensor. For driving

pressurized air valves a cluster of five MOSFET based static switches is used and for communicating with inverter, one RS485 serial interface is also included [3][4].

When some object arrives at the beginning of belt, the optical proximity sensor detect it, the processing unit send to inverter a "START" message frame and sorting process is started to.

After color analysis, each object will be blowed off to appropriate section of a clustered container. After each puff, controller transmits a data frame to the main control unit (if any), providing several information about color and number of analyzed pieces.

If no more pieces are detected on the conveyor belt, after a period of time, the control unit transmits to inverter a "STOP" message frame.

The main system components, providing motor control, color analysis and pieces sorting, are shown in figure 1.

The example allows remote connectivity between microcontroller based processing unit and a monitoring main unit (if any), using the same serial interface, or another one.

There are many development boards, suitable for hardware implementation, on marketplace. In this example, an ST Microelectronics' STM32 family ARM based platform with some other electronic components, are used.

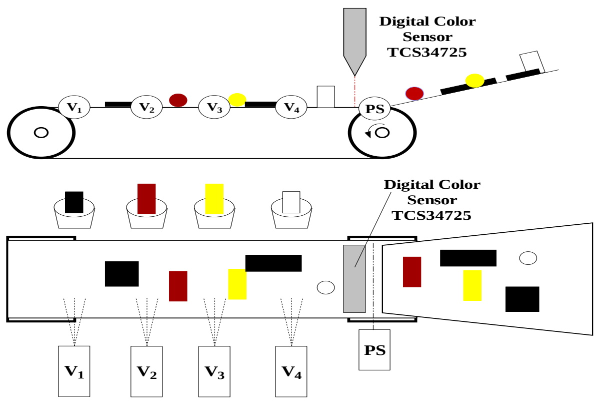

Fig. 1. The color based sorting system's main components

## 2. HARDWARE COMPONENTS

### 2.1. General description

The processing device has the following main parts, as shown in figure 2.

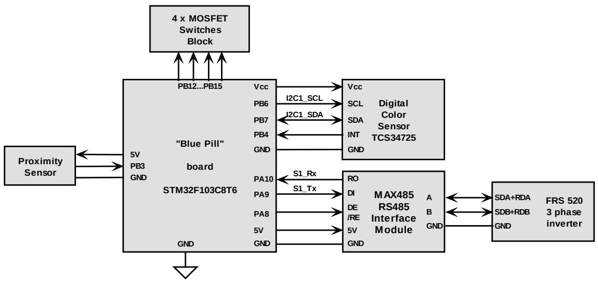

Fig. 2. The processing device hardware components

The system allows optical proximity detection, color measurements and analysis, by using digital inputs/outputs and digital smart sensor, besides I$^2$C serial communication bus.

First circuit is quite simple, involving circuits and simple routine only, by using optical curtain sensor. Color measurement is based on digital sensor, using I$^2$C serial communication interfaces.

For driving solenoids of air valves, a block of power MOSFET transistors are used.

Each component will be described, in details, in following chapters.

### 2.2. Processing Unit

As processing unit, one STM32F103C8 T6 microcontroller is being used. It is an ARM (32bit) type processor at 76 MHz with 64kB of flash and 20kB of RAM memory, several available A/D inputs, digital I/O, UART, I2C and SPI serial interfaces [5][6].

For this application, a "Blue Pill" development board well covers all exigencies and can be fitted into an appropriate DIP socket [7].

### 2.3. Color sensor

There are different sensor module types on the market place.

The TCS3472 device is a Color Light-to-Digital Converter with IR Filter. It provides a digital return of red, green, blue (RGB), and clear light sensing values.

The IR spectral component of the incoming light is minimized due to an IR blocking filter, integrated on-chip and localized to the color sensing photodiodes, allowing getting accurate color measurements [8].

As it has high sensitivity, wide dynamic range, and IR blocking filter, the TCS3472 is used in a wide range of applications, including RGB LED backlight control, solid-state lighting, health / fitness products, industrial process controls and medical diagnostic equipment, light color temperature measurement, fluid and gas analysis, product color verification and sorting.

The TCS3472 light-to-digital converter contains:

- a 3 × 4 photodiode array - composed of red-filtered, green-filtered, blue-filtered, and clear (unfiltered) photodiodes, coated with an IR-blocking filter;
- four analog-to-digital converters (ADC) that integrate the photodiode current; the four integrating ADCs simultaneously convert the amplified photodiode currents to a 16-bit digital value;
- data registers: upon completion of a conversion cycle, the results are transferred to the data registers, which are double-buffered to ensure the integrity of the data;
- a state machine, controlling all of the internal timing, as well as the low-power wait state;
- an I²C interface: communication of the TCS3472 data is accomplished over a fast, up to 400 kHz, two-wire I²C serial bus. The industry standard I²C bus facilitates easy, direct connection to microcontrollers and embedded processors

Fig. 3. TCS34725 sensor module with 4150K color temperature LED light Adafruit industries [11].

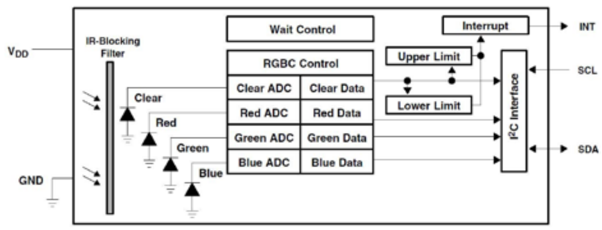

Fig. 4. TCS34725 color sensor – block diagram [8]

In addition to the I²C bus, the TCS3472 provides a separate interrupt signal output. When interrupts are enabled, and user-defined thresholds are exceeded, the active-low interrupt is asserted and remains asserted until it is cleared by the controller. This interrupt feature simplifies and improves the efficiency of the system software by eliminating the need to poll the TCS3472. The user can define the upper and lower interrupt thresholds and apply an interrupt persistence filter.

The interrupt persistence filter allows the user to define the number of consecutive out-of-threshold events necessary before generating an interrupt. The interrupt output is open-drain, so it can be wire-ORed with other devices.

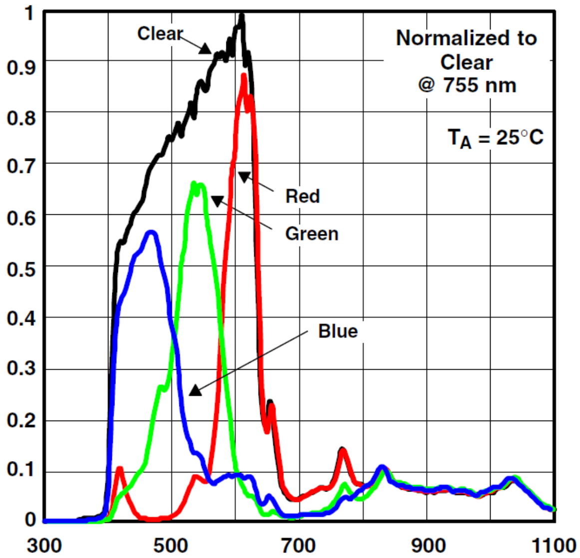


Fig. 5. TCS34725 color sensitivity [8]

The digital color sensors provide red, green, blue and clear channel outputs with an integrated IR filter over the RGBC channels. Clear refers to an open channel with no color filter over the sensor. The RGBC channels are red (RDATA - abbreviated as R), green (GDATA - abbreviated as G), blue (BDATA - abbreviated as B) and clear (CDATA- abbreviated as C) and are 16 bit values.

In the calculations, for some applications, the IR content is negligible and can be ignored. However, in applications that need to measure ambient light levels, incandescent lights and sunlight have strong IR contents. Most IR filters are imperfect and allow small amounts of residual IR to pass through. For IR intensive light sources, additional calculations are needed to remove the residual IR component [9].

The IR component can be calculated as follows:

$$IR = \frac{R+G+B}{2}$$

The IR compensated channels - labeled as R', G', B', C' – will be calculated using the formulas:

$$R' = R - IR$$

$$G' = G - IR$$

$$B' = B - IR$$

$$C' = C - IR$$

Color temperature **CT** (stated in degrees Kelvin) is linked to the glowing color of a piece of metal when heated to a particular temperature. The color temperature goes from red at lower temperatures to blue at higher temperatures [9][10].

A simple method to determine color temperature (CT) is to use the ratio of blue to red light, using the formula:

$$CT = CT\_Coef * \frac{B'}{R'} + CT\_Offset$$

IR cancellation is critical to the operation.

The coefficients are given in device's datasheets. For TCS3472, the above formula becomes:

$$CT = 3810 * \frac{B'}{R'} + 1391$$

### 2.4. RS485 multidrop serial interface converter

Due to inverter configuration, as an industrial device [14], a multidrop serial communication interface must be used. There are many types of transceivers that implements RS485, with two or four wires connection architecture. Because only half duplex transmission mode is used, two wires are enough to connect the processing unit with the inverter, as shown in figure 6.

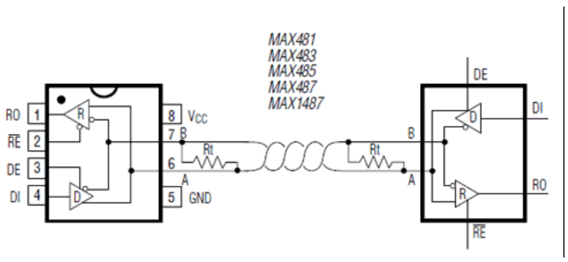


Fig. 6. MAX485 connection diagram [13]

### 2.5. Three phase variable frequency inverter

The electric drive motor is electrically supplied by using remote controlled frequency inverter, one Mitsubishi FR-S520 in this application [14]. The paper describes the communication messages for START/STOP procedures only, for other types of inverter those can be easily adapted.

It is necessary to prior set up several parameters, like station address, communication protocol, running speed and function mode - remote controlled over RS485 serial communication interface. After that, the inverter can be started and stopped remotely, using messages sent by processing unit.

The following communication parameters can be set using buttons and rotary encoder on inverter front panel:

n1 = 1 Station number
n2 = 192 Communication speed 19200 bps
n3 = 0 Stop bit + length 8 data 1 stop
n4 = 0 Parity check
n5 = 3 Number of communication retries
n6 = 1 Communication check time interval (s)
n7 = 20 Wait time setting

n8 = 0 Operation command write

n10 = 1 Link start mode selection (Computer link)

n11 = 2 CR/LF selection (CR+LF)

“START” and “STOP” messages are sent by using protocol A’ format, as is sown bellow.

Start message

| Enq | Addr | Func | Data | Sum | CR+LF |
|---|---|---|---|---|---|
| 0x05 | 0x01 | 0xFA | 0x12 | 0x14B | 0x0D0x0A |
| 05 | 30 31 | 46 41 | 31 32 | 34 42 | 0D 0A |

Stop message

| Enq | Addr | Func | Data | Sum | CR+LF |
|---|---|---|---|---|---|
| 0x05 | 0x01 | 0xFA | 0x00 | 0x14B | 0x0D0x0A |
| 05 | 30 31 | 46 41 | 30 30 | 34 38 | 0D 0A |

The messages are ASCII characters string, starting with 0x05 (ENQ – enquiry), station address (0-31 in range), function codes, data, check sum and terminator delimiter field (CR – carriage return and LF – line feed). As noted before, station address was set to 1. Function code 0xFA represents “write RUN mode” and can control various rotation speeds and directions. Data field with a value of 0x12 (binary 00010010) means “start forward at medium speed”, configured before by using parameter setting, and a value equal to 0x00 means “STOP”.

Address, function and data fields are represented in ASCII manner, so each byte means two characters in length.

The check sum field is compute by using arithmetic summation of each character value between, (but without) start and end delimiters [14].

### 2.6. MOSFET static switches block

Four independent static switches are used to drive air valves. Functions are implemented directly in microcontroller program code, each one being treated as simple digital output. If IRF520 type (which threshold is greater than 4V) is directly connected to microcontroller, 5 Volt tolerant ports, or level shifter circuits, must be used [12].

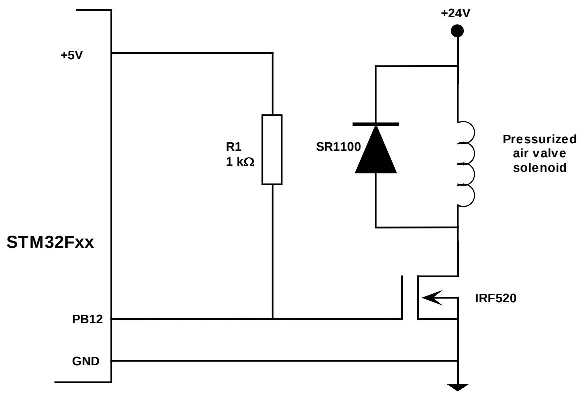


Fig. 7. MOSFET based static switch

A simple trick is shown in figure 7, where the microcontroller’s output pins must be five volt tolerant and configured as open drain.

### 2.7. How these blocks work together?

All components described above are connected as shown in figure 2.

When proximity sensor detects a piece on the beginning of conveyor belt, it changes its logic level. At this moment microcontroller send a “Start” message to inverter that starts to supply electric motor at a prior defined speed.

When a piece arrives under the color sensor, the analyzing process starts and sets an appropriate delay time for the moment when that piece must be blown off to appropriate container. In the same time, the process unit send to the main control unit (if any) a message with some information about current analyzed piece.

Depending on color, each piece will have different delay time, calculated by using belt speed and distance between containers.

When no more piece are loaded on the belt, after a period of time, the processing unit send a “Stop” message to inverter, ending the sorting process.

## 3. SOFTWARE COMPONENTS

### 3.1. Main C Code routines of processing unit

The code includes libraries from STM32duino open source codes, or public domain repositories. Most of them are used in original version, but few are modified by the author to reduce the amount of memory usage. The original code was developed in unsightly raw guise, but it is rewrite for better understanding. It was tested in a real sorting system project since date indicated in the main code header.

```
/**********************************************
Title:
Recyclable_Materials_Sorting_System.ino
Author: Florin Dragan  florind@mas.utcluj.ro
Date Created: 25-10-2018
Last Modified: 15-11-2018
Thanks to KTOWN (Adafruit Industries) for
color sensor library, Roger Clark for porting
many usefull STM32 libraries and to many
others, for their contribution on developing
libraries used in this project.
**********************************************/
```

To develop the application many libraries are used, but the following libraries must be explicitly included:

```
//I2C two wires communocation interface;
#include <Wire.h>
//Digital color sensor APIs
#include <Adafruit_TCS34725.h>
/* Initialise with default values (int time =
2.4ms, gain = 1x) */
Adafruit_TCS34725 tcs = Adafruit_TCS34725();
```

The following pins are used for digital input, output, or for serial communication and must be 5V tolerant to enable compatibility with other used components:

PB3 input from proximity sensor
PB12 – PB15 output to control MOSFET switches
PB6 – SCL I2C1 serial interface
PB7 – SDA I2C1 serial interface
PB4 – INT input from TCS34725 interrupt pin
PA10 – Rx input of Serial 1 interface -> RS485
PA9 – Tx output of Serial 1 interface -> RS485
PA8 – RTS output used to control RS485 transmission

Several parameters must be defined first:

```
/*Delay advance time to selection representing the necessary time to walk between color sensor and appropriate recipient. Depends on conveyor belt speed and distances*/
#define A 820 // Color temperature A
#define A 1200 // Color temperature B
. . .  and so on

const unsigned int delay_A = 1000;
const unsigned int delay_B = 1500;
const unsigned int delay_C = 2000;
const unsigned int delay_D = 2500;
const unsigned int pulse_time = 200;
const unsigned int baud_rate = 19200;

/* Start and stop message strings */
const char start_msg[] =
{0x05,0x30,0x31,0x46,0x41,0x31,0x32,0x34,0x42,
0x0D,0x0A};
const char stop_msg[] =
{0x05,0x30,0x31,0x46,0x41,0x30,0x30,0x34,0x38,
0x0D,0x0A};

/*Variable used to define the moment when different piece reach the color sensor, one for each cathegory*/
unsigned int moment_A;
unsigned int moment_B;
unsigned int moment_C;
unsigned int moment_D;
unsigned int moment_run;

/*Time gap between chars and frames*/
unsigned int t_15, t_35;

/* Flags used to define some events*/
boolean selection_flag_A = false;
boolean selection_flag_B = false;
boolean selection_flag_C = false;
boolean selection_flag_D = false;

/*The initialisation sequence (brief)*/

void setup(void) {
 Serial1.begin(baud_rate); //RS485 interface
 if (baud > 19200) {
    t_15 = 750;
    t_35 = 1750;
 } else {
    t_15 = 15000000/baud_rate;
    t_35 = 35000000/baud_rate;
 }
```

Digital pins must be configured according with their functions.

Digital input pins generally use internal pull-up or pull-down resistors.

Digital output pins, that control MOSFET static switches, must be configured as "open drain" allowing connection to a different voltage (5V in this case) by using an external pull-up resistor.

```
/*Pins used as input*/
pinMode(PB3, INPUT); // from proximity sensor
pinMode(PB4, INPUT_PULLUP); // INT TCS34725
/* The following pins are used to control switches and use pull up resistor to 5V, for better function of MOSFET channel and must be logic LOW*/
pinMode(PB12, OUTPUT_OPEN_DRAIN);
digitalWrite(PB12, LOW);
. . . .

tcs.begin();
. . .
} // end setup
```

The following function is used to send message on RS485 serial interface, and controls the "transmit enable pin of MAX485 transceiver chip. The pin level is raised before sending data and lowered after awaiting for serial interface to complete the transmission.

```
void sendRS485Frame(unsigned char frame,
unsigned char bufferSize) {

digitalWrite(PA8, HIGH);
for (unsigned char i = 0; i < bufferSize; i++)
Serial1.write(frame[i]);

/* For STM32F103
    usart_reg_map *regs = USART2->regs;
/* Wait for Transmission Complete */
    while ( !(regs->SR & USART_SR_TC) );
/* Allow a frame delay to indicate End of Transmission */
     delayMicroseconds(T3_5);
     digitalWrite(PA8, LOW);
/* Empty receive buffer */
if (Serial1.available()) {
  while(Serial1.available()) Serial1.read();
}
```

The processing routine operates as "state machine" tasks using several flags to implement "flip-flop" functions.

```
void loop(void) {

static boolean send_flag = true;
static boolean stop_flag = false;
static boolean start_flag = false;

/* Check the proximity sensor status, first for START and STOP routine*/
if (digitalRead(PB3)) {
 start_flag = true;
 moment_run = millis()/1000;
}

if (start_flag && send_flag) {
send_mes_to_inverter(start_msg, 11);
send_flag = false;
}

if(moment_run > (millis()/1000) + run_time) {
```

```
moment_run = (millis()/1000);
 if(start_flag == true){
  start_flag = false;
 } else {
  if(send_flag && !stop_flag){
  send_mes_to_inverter(stop_msg, 11);
  send_flag = false;
  stop_flag = true;
  start_flag = true;
 }
}
/* Variables used for color analyse*/
uint16_t r, g, b, c, colorTemp, lux;
```

When a piece arrives under the color sensor, this transmits an interrupt signal and the processing unit starts to acquire the raw values of color components, reading them from TCS34725 store registers. After that, several calculation functions, according with DN40 recommendation, are used.

```
if(digitalRead(PB4)) { proc_flag = true;
last_proc_flag = proc_flag;
} else { proc_flag = false;}

if ((prog_flag!=last_prog_flag)&&prog_flag) {
tcs.getRawData(&r, &g, &b, &c);
// colorTemp =
tcs.calculateColorTemperature(r, g, b);
colorTemp =
tcs.calculateColorTemperature_dn40(r, g, b,
c);
```

Comparing color temperature with suitable range defines which air valve coil to be activated, according to sorting category, A, B, C and so on

```
 if (colorTemp < A) {
   moment_A = millis();
   selection_flag_A = HIGH;
}
  if (selection_flag_A) {
  unsigned int temp_A = moment_A + delay_A;
  if (temp_A >= milis()) {
digitalWrite(pin_A, HIGH);
  }
  if (temp_A >= milis() + pulse_time) {
   digitalWrite(pin_A, LOW);
  selection_flag_A = LOW;
   }
  }  //end if (selection_flag_A)
. . .  other selection cases
```

## 4. CONCLUSIONS

This paper describes a solution developed for color based sorting process of recyclable materials obtained by disassembling components used in IT&T devices. Method applys color analysis using digital color sensor TCS34725 made by **ams AG, Tobelbader Strasse 30, 8141 Premstaetten, Austria**. The company produces a large variety of optical sensors for diverse applications. By converting recyclable material color in color temperature simplify the analysis and make sorting process easyer to implement.

The paper describes a cheap solution that uses an ARM microcontroller base processing unit that can work as stand alone system, or as a part of a more complex Recyclable Materials Management System.

## 5. ACKNOWLEDGMENTS

This paper work succeeds a research activity, as part on recyclable materials sorting system solutions, that was financial supported as a grant of the Romanian Ministry of Research and Innovation, CCCDI-UEFISCDI, project number PN-III-P1-1.2-PCCDI-2017-0652 / 84PCCDI/2018, within PNCDI III.

## REFERENCES


1. K. Feldmann, H. Scheller, The Printed Circuit Board - A Challenge for Automated Disassembly and for the design of Recyclable Interconnect Devices, International Conference on Clean Electronics Products and Technology, Edinburgh, 1995.
2. Feldmann, K.; Scheller, H, Partially automated conceptions for seperation of used electronic devices, VDIWDE-Gesellschaft Mikro- und Feinwerktechnik, Seminar 'Praxis of electronic scrap recycling', Duisburg, 1993.
3. Jiu Huang, Thomas Pretz, Zhengfu Bian Intelligent Solid Waste Processing Using Optical Sensor Based Sorting Technology 2010 3rd International Congress on Image and Signal Processing (CISP2010)].
4. Timothy Henry. Laurence, Ishak, Ferry Jie, Design and Construction of Color Sensor Based Optical Sorting Machine. 2017 5th International Conference on Instrumentation, Control, and Automation (ICA) Yogyakarta, Indonesia, August 9-11, 2017.
5. STMicroelectronics, RM0008, Reference manual STM32F101xx, STM32F102xx, STM32F103xx, STM32F105xx and STM32F107xx advanced ARM®-based 32-bit MCUs, DocID13902 Rev 16, 2015.
6. STMicroelectronics, STM32F103x8, STM32F103xB, Medium-density performance line ARM®-based 32-bit MCU with 64 or 128 KB Flash, USB, CAN, 7 timers, 2 ADCs, 9 com. interfaces, DocID13587 Rev 17, 2015.
7. https://www.techshopbd.com/uploads/product_document/STM32bluepillarduinoguide(1).pdf
8. ams, TCS3472 Color Light-to-Digital Converter with IR Filter ams Datasheet v1-03] 2018-Mar-14.
9. ams, Lux and CCT Calculations using ams Color Sensors, Application Note: DN40-Revision 1.0 / 26/08/13
10. ams, Kerry Glover, Intelligent Opto Sensor, Designer's Notebook, Using the Lux Equation, Number 29A, August 2011
11. Bill Earl, Adafruit Color Sensors Adafruit Industries Last updated on 2018-11-05 03:48:12 PM UTC
12. Vishay Siliconix, IRF520, SiHF520 Power MOSFET, Document Number: 91017 S11-0511-Rev. B, 21-Mar-11.
13. Maxim Integrated, Low-Power, Slew-Rate-Limited RS-485/RS-422 Transceivers, 19-0122; Rev 10; 9/14
14. Mitsubishi Electric, FRS-500 Frequency Inverter instruction manual (detailed),


Florin Drăgan
Faculty of Electrical Engineering, Technical University of Cluj-Napoca, 26-28, G. Barițiu st., Cluj-Napoca, Romania
Florin.Dragan@ethm.utcluj.ro

Rodica Holonec,
Faculty of Electrical Engineering, Technical University of Cluj-Napoca, 26-28, G. Barițiu st., Cluj-Napoca, Romania
Rodica.Holonec@ethm.utcluj.ro

Romul Copîndean,
Faculty of Electrical Engineering, Technical University of Cluj-Napoca, 26-28, G. Barițiu st., Cluj-Napoca, Romania
Romul.Copindean@ethm.utcluj.ro