Design and implementation of the driver system for a Hamamatsu C12880MA microspectrometer

Patryk Sokołowski, Maciej S. Wróbel, Katarzyna Karpienko[*]

Faculty of Electronics, Telecommunications and Informatics, Gdańsk University of Technology, G. Narutowicza 11/12, 80-233 Gdańsk,

Abstract—Recent miniaturization developments in devices for spectroscopy have reduced greatly their costs and increased their availability for a wide range of users and applications. This paper presents the design and implementation of a driver system for a Hamamatsu C12880MA microspectrometer. The system implementation was carried out and compared using two independent microcontroller modules: Arduino Uno and STM32F411RE Nucleo. We assessed the microspectrometer system parameters like resolution and integration time, as influenced by the choice of a driver microcontroller. We present a possible application of this microspectrometer system for measurements of ground coffee beans reflectance for control of coffee roasting.

Spectroscopy is a powerful method for chemical analysis of samples. An immense variety of objects, from biological and medical samples, through foods and agricultural produce to technical and industrial materials can be analyzed with the use of spectroscopy. The main aspect of a spectroscopic device, regardless of a specific spectral method, is the division of electromagnetic radiation into separate wavelengths and their detection. The advancement in miniaturization of spectrometers [1] enables a wider use of this powerful technique in practical applications due to cost reduction and accessibility.

This paper presents the design and implementation of a driver circuit for the C12880MA Hamamatsu microspectrometer. The driver circuits are required to record the electrical output of a microspectrometer, for further processing. Two different microcontrollers: Arduino Uno and STM32F411RE Nucleo were used in driver systems. It was necessary to generate clock signals and control pulses for spectrometer output and controlling the integration time. The driver was tested and validated. The schematic of the system with a driver circuit and a microspectrometer is presented in Fig. 1. The design of the driver circuit required the provision of power voltage, microspectrometer clock frequency, START signal and sampling of an analog spectrometer output signal.

The C12880MA Hamamatsu microspectrometer chassis is only $20.1 \times 12.5 \times 10.1$ mm. A concave reflective grating is used to separate wavelengths of incoming optical radiation, in the range $340 - 850$ nm, and images it onto a CMOS matrix detector with 288 pixels [2].

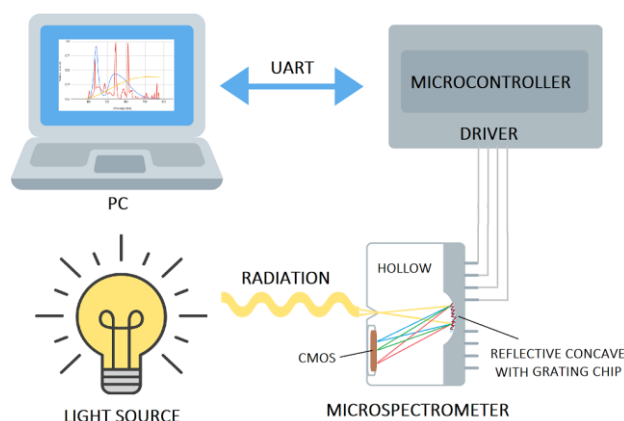[*] E-mail: katkarpi@pg.edu.pl

Fig. 1. Schematic of the spectra measurement system with a driver circuit and a Hamamatsu C12880MA microspectrometer.

There is wide availability of microcontrollers that fulfill the requirements of the driver circuit. The devices were chosen due to their price and ease of application writing. Arduino UNO is based on AVR ATmega328 clocked at 16 MHz and has 14 I/O (6 can be used for PWM and the other 6 as analog), a 10-bit ADC. The power and communication can be supplied concurrently through USB [3]. Nucleo STM32F411RE is based on 32-bit Cortex M4 clocked at 100 MHz. It shares the pinout with Arduino but uses a 12-bit ADC. Communication capabilities are: SPIE, UART, I^2C, and USB 2.0 [4]. The main difference is the resolution of ADC, and clock frequency: 10 bits and 135 kHz, and 12 bits and 505 KHz.

The designed driver circuit utilizes three analog I/O pins: to provide clock signal, generation of control impulses, and to receive the video signal from the microspectrometer. The video signal carries the analog values of voltage from microspectrometer' ADC, relaying the intensity at consecutive pixels of the CMOS matrix detector. The block diagram of the program controlling the driver is presented in Fig. 2. The program is responsible for setting up spectrometer integration time and capturing video signal.
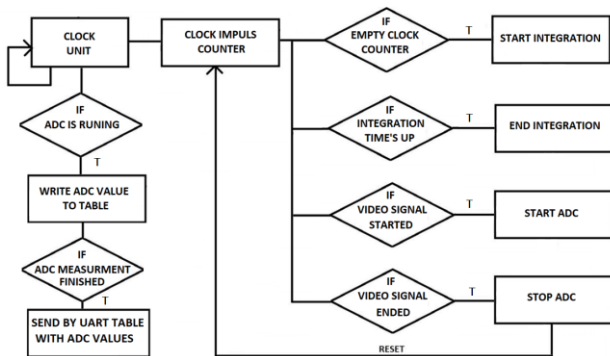
Fig. 2. Driver code block diagram.

Figure 3 presents the signals obtained by connecting the oscilloscope to the microspectrometer analog output video signal (blue line), which carries the pixel values from the CMOS matrix, and the spectrometer clock signal (violet line) generated by the microcontroller. At the beginning we can notice higher density of impulses related to the integration time setup procedure.
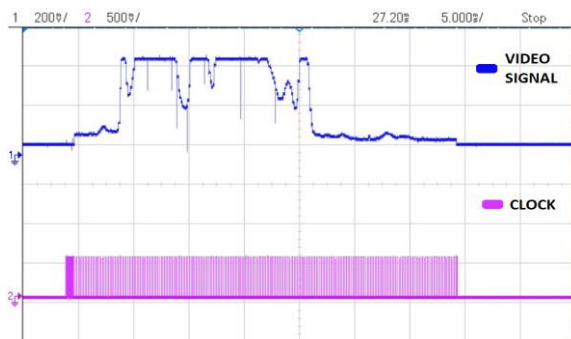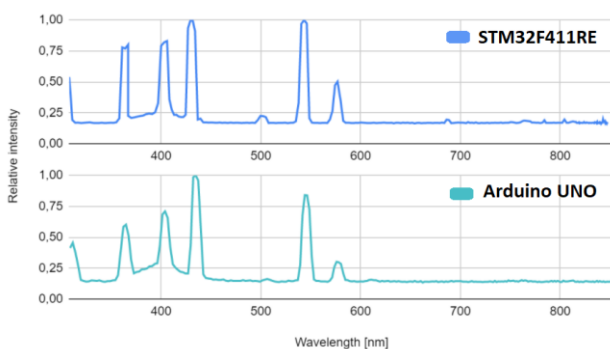

Fig. 3. Video signal and clock timing.

Fig. 4. Calibration lamp measurements.

Calibration of the spectrometer was performed with the Hg-Ar calibration lamp (CAL - 2000, Ocean Optics) which emits the mercury and argon emission lines at 313, 356, 404, 435, 546, 576 and 579 nm with about 1 nm bandwidth. The spectrometer resolution was assessed by comparison of the recorded spectrum and known line

positions and bandwidths. The resolution is about 11 nm, which is about the producer declared resolution of 15 nm. Additionally, the spectral range is broader than in the specifications [2] and spans from $309 - 868$ nm. Figure 4. presents the recorded calibration spectrum for both microspectrometer systems.

The set-up was tested by measured emission of common light sources. The reference spectra were recorded by a Konica Minolta CS-200 spectrometer. The comparison of light source spectra obtained by the microspectrometer with different microcontrollers in the driver system and the reference are presented in figures 5 – 7, presenting a white LED, a CFT, and an incandescent light bulb, respectively.
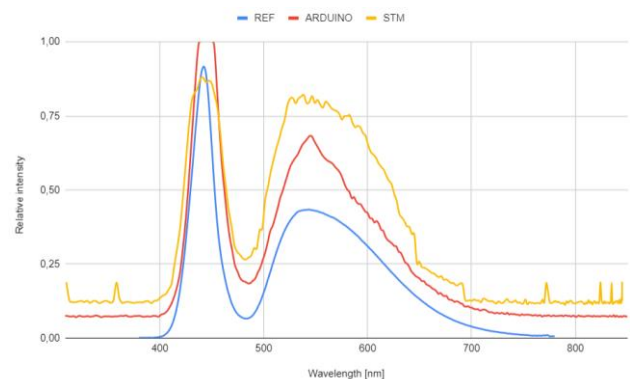

Fig. 5. Spectrum of white LED.

The characteristics of the spectra recorded by the microspectrometer is similar to the spectra from the reference spectrometer, with an exception of incandescent measurement, where the values read in the near infrared area decrease. This is due to lower sensitivity of a detector in this area. A similar decrease was observed with an earlier measurement of the calibration lamp.

There are differences in recorded spectra, with the STM having more noisy spectra in general, with slight shifts of the peaks relative to the reference, which does not occur when Arduino is used. This may be related to some timing issues or handling of the ADC. Nevertheless, the spectra are clearly distinguishable and with little optimization of integration time and number of accumulations or averaging can be used for further analysis. Thus, we confirm the proper handling of a video signal by the driver circuit, regardless of microcontroller choice.
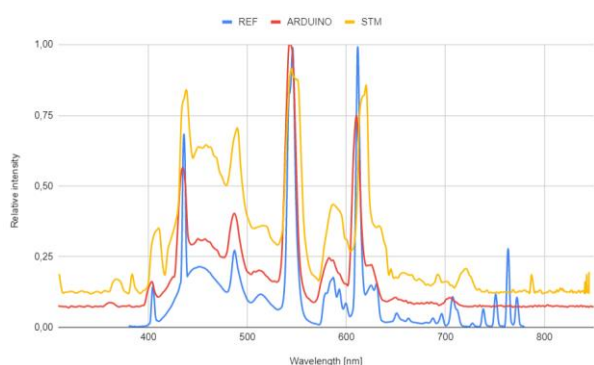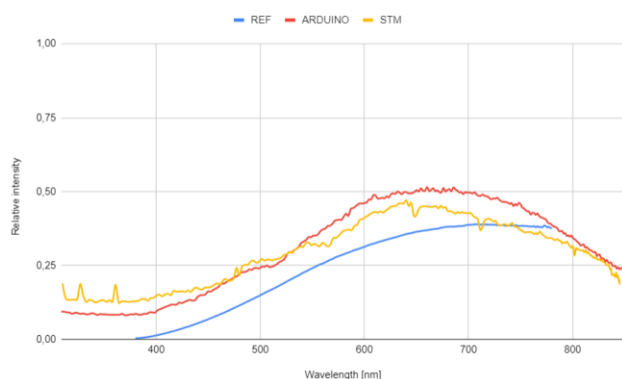
Fig. 6. Spectrum of CFR source.

Fig. 7. Spectrum of incandescent light bulb.

Here we present an application of the implemented system on samples of ground coffee beans, obtained at different stages of the roasting process. Such a process is normally controlled by a various number of sensors, such as heating-timing profile, composition of exhaust gasses, but the most common and traditional way is visual observation of the Millard reaction causing the beans to go from white to dark brown [5].

We have tested the system on three different levels of roasting, samples 1 – 3 respectively (shown in Figure 8). The beans were ground and uniformly dispersed on a plate in a dark chamber with a white LED as a light source. The reflection spectra of samples were measured by the system. Coffee roasting spectra are shown in Figure 9 for the controller built on the Arduino (upper), and for the STM32F411RE driver (lower), where the lines indicate the sample number (yellow - sample 1, red - sample 2, blue - sample 3). The integration time for the STM driver is 1.383 µs, and for Arduino 1.75 µs. The integration time was selected experimentally so as to ensure a high signal level while not allowing the detector to become oversaturated.

The results indicate that coffee, depending on the level of roasting, differed mostly in the intensity of the red color (wavelength approx. 630 nm). This was true for both systems despite small differences in intensities and noise. This confirms the assumption that spectral measurements can be successfully performed with the constructed system.

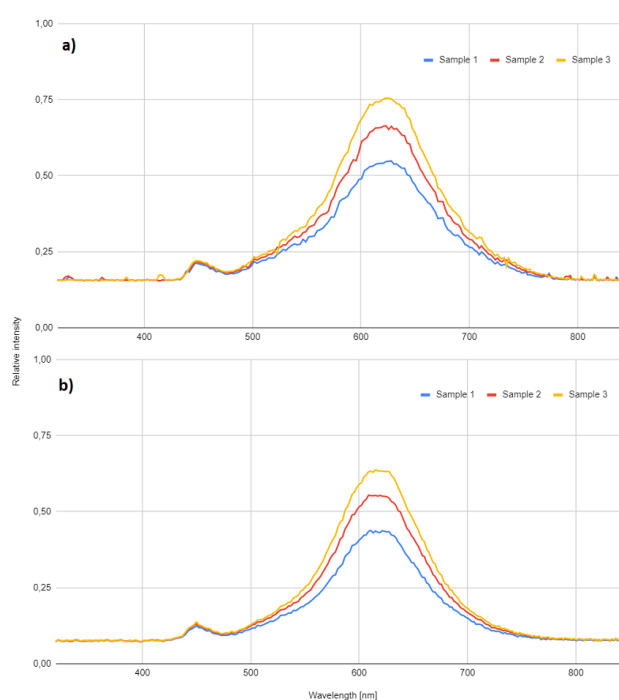Fig. 8. Ground coffee beans samples at various degrees of roasting.

Fig. 9. Coffee roasting spectra. Driver based on a) STM, b) Arduino.

References

[1] R.A. Crocombe, Appl. Spectr. **72**(12):1701(2018). doi:10.1177/0003702818809719
[2] Datasheet Hamamastu C12880MA
[3] Datasheet Arduino Uno
[4] Datasheet STM32F411RE
[5] J.A. Hernández, B. Heyd, G. Trystram, J. Food Eng. **89**.2, 156 (2008).