

STEROWNIK DIOD LED RGB MOCY

Daniel PĘCZKOWSKI¹, Krystyna Maria NOGA²

¹ Dyplomant Wydziału Elektrycznego, Akademia Morska w Gdyni

² Akademia Morska w Gdyni, Katedra Automatyki Okrętowej
tel: 58 55 86 471, e-mail: k.noga@we.am.gdynia.pl

Streszczenie: W artykule został przedstawiony układ automatycznego sterowania diodami elektroluminescencyjnymi, które emitują światło o podstawowych barwach, tj. czerwone, zielone i niebieskie. Uzyskana jasność i kolor oświetlenia odpowiadają wielkościom zadany przez opracowaną specjalnie na potrzeby sterownika aplikację komputerową.

Słowa kluczowe: pomiar koloru, sterownik LED RGB, mikrokontroler.

1. WSTĘP

Urządzenia wykorzystywane do pomiaru koloru nie mierzą go bezpośrednio. Czujniki natężenia oświetlenia oraz barw, które wykorzystują fotoelementy, pozwalają jedynie na określenie ilości światła padającego na ich powierzchnię. Określenie koloru światła padającego na czujnik wymaga zastosowania odpowiednich przyrządów lub filtrów optycznych, które pozwalają rozdzielić badane światło na poszczególne składowe. Najczęściej stosowane są czujniki światła oparte o fotodiody [1, 2].

W Katedrze Automatyki Okrętowej Akademii Morskiej w Gdyni, w ramach pracy dyplomowej inżynierskiej [3], zaprojektowano i skonstruowano układ sterujący diodami elektroluminescencyjnymi, które emitują światło o podstawowych barwach, tj. czerwone, zielone i niebieskie. Dzięki zastosowaniu przetwornika koloru w pętli automatycznej regulacji uzyskiwana jasność i kolor oświetlenia odpowiadają wielkościom zadany. Oprogramowanie sterownika umożliwia ponadto komunikację z komputerem PC w celu zmiany parametrów. Manualne pomiary przetwornikiem koloru zostały wykorzystane do określenia charakterystyk tego przetwornika i strumienia świetlnego diod LED. Ponadto pomiary, wykonane przy pomocy aparatu cyfrowego i diod LED, zastosowano do wyznaczenia punktu bieli oraz wyznaczenia wzorców kolorów dla przetwornika.

2. PRZEGLĄD ROZWIĄZAŃ STEROWNIKÓW LED RGB

Tematyka sterowania diodami LED obejmuje wiele różnych rozwiązań, począwszy od najprostszych układów ściemniaczy, a kończąc na zdalnie sterowanych lampach estradowych i wyświetlaczach matrycowych. Najprostsze konstrukcje sterowników LED RGB nie posiadają układów do automatycznej regulacji emitowanego światła przy pomocy przetwornika. Zmiana natężenia oświetlenia oraz jego koloru odbywa się najczęściej poprzez wykorzystanie pilota

zdalnego sterowania, aplikację komputerową lub panel naścienny. Sterowniki tego typu produkowane są przez wiele firm, np. QLT, która oferuje ich szeroki wybór [4]. W praktyce, np. do oświetlenia sceny, często wymagane jest zastosowanie lamp o zróżnicowanych kolorach. Zastosowanie w tym celu reflektorów LED jest znacznie bardziej energooszczędne oraz mniejsze gabarytowo w porównaniu do klasycznych rozwiązań z wykorzystaniem lamp żarzeniowych lub żarowych. Sterowanie diodami LED zwykle odbywa się poprzez wykorzystanie protokołu DMX512 [5]. Innym rozwiązaniem jest system oświetlenia zaprezentowany na stronie internetowej firmy Digi-Key [6]. Umożliwia on regulację temperatury światła białego oraz automatyczną regulację natężenia oświetlenia w zależności od natężenia światła dziennego. Redukcja jasności diod LED pozwala na obniżenie zużycia energii w słoneczne dni. W celu automatycznej regulacji naświetlenia pomieszczenia wykorzystano przetwornik światła dziennego, który służy do pomiaru światła przychodzącego z zewnątrz. Ponadto do pomiaru światła wewnątrz pomieszczenia, w celu automatycznej regulacji punktu bieli, zastosowano przetwornik koloru.

3. BUDOWA STEROWNIKA

Skonstruowany sterownik LED RGB spełnia następujące założenia projektowe:

- zasilanie układu sterowania odbywa się ze źródła napięcia stałego lub poprzez port USB,
- sterowane źródło zasilania diod LED mocy pracuje poprawnie przy zasilaniu napięciem stałym w zakresie 9-24 V, ponadto zapewnia regulację jasności,
- mikrokontroler (MCU, ang. *Microcontroller*) ATSAM D21G18 firmy Atmel umożliwia operację na liczbach o długości przynajmniej odpowiadającej liczbom odczytywanym z przetwornika i wykorzystywanym do zmian nastaw wypełnienia sygnału PWM sterującego diodami,
- komunikacja urządzenia z PC poprzez port USB zapewnia wymianę danych, sterowanie diodami, konfigurację układu i zmiany nastaw,
- opracowane aplikacje komputerowe do sterowania ręcznego diodami LED z odczytem danych z sensora koloru oraz do sterowania automatycznego kolorem i natężeniem światła emitowanego przez diody LED umożliwiają badanie sterownika oraz jego użytkowanie,

- sterowanie diodami LED mocy odbywa się zarówno w układzie regulacji automatycznej oraz układzie sterowania ręcznego,
- algorytm sterujący diodami może być zapisany w pamięci mikrokontrolera lub pamięci komputera PC,
- zaimplementowano wzorce kolorów podstawowych, złożonych oraz bieli,
- jako źródło światła wykorzystano moduły LED mocy zawierające emiterzy światła o kolorach: czerwonym, zielonym i niebieskim.

Skonstruowany sterownik LED umożliwia pracę w trybach przedstawionych w tabeli 1, przy czym w kolumnach a, b, c przedstawiono informacje dotyczące odpowiednio: obsługi przetwornika koloru, generatora PWM, protokołu DMX512.

Tabela 1. Tryby pracy sterownika

tryb pracy Mode	sterowanie LED'ami poprzez	obsługa		
		a	b	c
USB_S	USB	X		
USB_S_DMXX	USB			X
USB_S_DMXX_S	USB	X		X
USB_S_PWM	USB		X	
USB_S_PWM_S	USB	X	X	
USB_DMXX_PWM	USB		X	X
USB_DMXX_PWM_S	USB	X	X	X
DMXX	MCU			X
DMXX_S	MCU	X		X
PWM	MCU		X	
PWM_S	MCU	X	X	
DMXX_PWM	MCU		X	X
DMXX_PWM_S	MCU	X	X	X
IDLE - tryb specjalny	oczekuje na przejście w inny tryb			
REBOOT - tryb specjalny	automatycznie restartuje sterownik			
RESTART_USB - tryb specjalny	restartuje połączenie USB komputera ze sterownikiem			
CONFIG - tryb specjalny	umożliwia konfigurację sterownika poprzez USB, np. zmianę domyślnego trybu pracy.			

Do budowy urządzenia wykorzystano mikrokontroler ATSAM21G18, który umieszczony jest na platformie *Arduino Zero Pro* [7]. Mikrokontroler o 32-bitowej architekturze ARM Cortex-M0+ wybrano ze względu na możliwość zakupu wraz z programatorem i debuggerem na jednym PCB (ang. *Printed Circuit Board*). Mikrokontroler posiada moduły komunikacji szeregowy SERCOM (ang. *Serial Communication Interface*), które obsługują wiele standardów komunikacyjnych. Jednym z nich jest standard I²C (ang. *Inter-Integrated Circuit*), który używany jest przez inne układy scalone zastosowane w zbudowanym sterowniku.

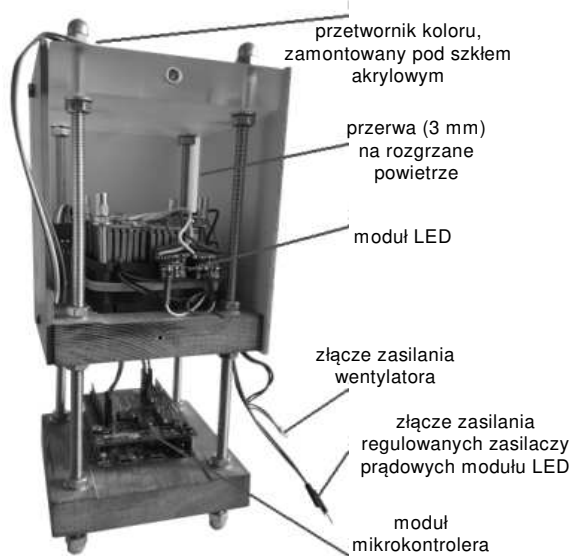
Przy doborze przetwornika światła TCS34725 firmy TAOS [1] uwzględniono między innymi koszt zakupu, dokładność odczytywanych wyników, sposób prowadzenia pomiaru, metodę komunikacji oraz dostępność. Przetwornik ten posiada matrycę dwunastu fotodiod pokrytą filtrami podczerwieni i kolorów, przy czym do pomiarów poszczególnych pasm RGB są wykorzystywane po trzy fotodiody, a pozostałe znajdują się wyłącznie pod filtrem podczerwieni. Przetwornik komunikuje się z mikrokontrolerem poprzez magistralę I²C, przez którą przesyłane są dane cyfrowe reprezentujące wyniki pomiaru światła poszczególnych pasm oraz wynik pomiaru światła widzialnego. Dane te są 16 bitowymi liczbami całkowitymi bez znaku. Układ przetwor-

nika koloru został zamontowany na PCB *Adafruit Flora Color Sensor with White Illumination LED* [8].

Do zasilania diod zastosowanych w zbudowanym sterowniku wykorzystano impulsowe przekształtniki prądu stałego o topologii buck, które obniżają napięcie. Przekształtniki te umożliwiają pracę z napięciami przekraczającymi napięcie przewodzenia diod. Układ scalony PT4115 firmy PowTech, wykorzystany w tych przekształtnikach, umożliwia sterownie sygnałem PWM oraz analogowym [9]. Do budowy sterownika LED zastosowano gotowe moduły oparte o układ PT4115, przy czym wprowadzono następujące zmiany: usunięto mostki prostownicze (układy zostały zasilone napięciem stałym), wyprowadzono przewód od pinu sterującego zasilaczem. Do budowy urządzenia wykorzystano także generator przebiegów PWM, który jest sterowany poprzez magistralę I²C. Wybrany układ scalony PCA9685 firmy NXP posiada 16 niezależnych wyjść o rozdzielczości 12 bitów [10]. Taktowanie tego układu zapewnia wbudowany oscylator RC. Częstotliwość nośną PWM można zmieniać poprzez wbudowany prescaler. Generator PWM jest umieszczony na PCB *Adafruit 16-Channel 12-bit PWM* [11].

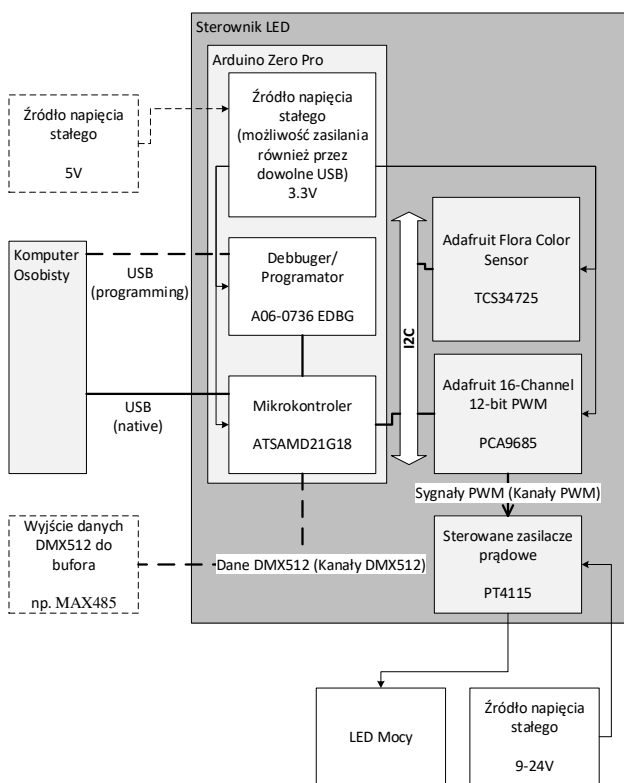
Do poprawnej pracy sterownik LED RGB wymaga komunikacji pomiędzy zastosowanymi układami scalonymi, do czego wykorzystana jest magistrala I²C. Natomiast do komunikacji z komputerem wykorzystano port USB skonfigurowany po stronie mikrokontrolera jako CDC (ang. *Communication Device Class*). W celu zwiększenia ilości kanałów PWM dostępnych dla mikrokontrolera został zastosowany układ generatora PWM typu PCA9685, co również zwiększyło rozdzielczość sygnału PWM. W zbudowanym sterowniku wykorzystano część kanałów PWM (tj. 12, 13, 14, 15), pozostałe mogą posłużyć do podłączenia większej ilości sterowalnych zasilaczy LED.

Do regulacji prądu skutecznego diod regulowane zasilacze prądowe korzystają z wygenerowanego sygnału PWM. Przetwornik światła wykorzystuje siatkę fotodiod przesłoniętych odpowiednimi filtrami, usuwającymi podczerwień, tak, aby możliwy był pomiar zawartości składowych RGB oraz natężenia światła. Dane uzyskane z przetwornika są przesyłane do mikrokontrolera, który nimi zarządza w zależności od wybranego trybu pracy. Przetwornik ustawiono w taki sposób, aby światło diod LED padało prostopadle na jego powierzchnię. Uzyskano w ten sposób możliwość wykonania pomiaru jasności diod LED, wyniki są wykorzystywane podczas automatycznej regulacji światła emitowanego. Algorytm regulacji może być wykonywany zarówno przez procesor mikrokontrolera jak i komputera PC, zależnie od wybranego trybu pracy. Dodatkową funkcją realizowaną przez sterownik LED RGB jest obsługa protokołu DMX512, który jest standardem wykorzystywanym w sterowaniu oświetleniem i innymi efektami scenicznymi. Na rysunku 1 przedstawiono konstrukcję mechaniczną przed końcowym montażem. Przetwornik koloru znajduje się na górze konstrukcji i skierowany jest na szkło akrylowe. Moduł LED wraz z układem chłodzenia oraz zasilaczami prądowymi znajduje się w środkowej części sterownika. Najniższa płyta jest wykorzystana do montażu modułu mikrokontrolera wraz z generatorem PWM. Przerwy pomiędzy płytami pilśniowymi umożliwiają przepływ powietrza.



Rys. 1. Sterownik diod LED RGB mocy

Schemat blokowy wykonanego sterownika wraz z jego peryferiami przedstawiono na rysunku 2.



Rys. 2. Schemat blokowy sterownika

4. KOMUNIKACJA PC ZE STEROWNIKIEM

Układ sterownika komunikuje się z komputerem poprzez port USB. Program zawarty w mikrokontrolerze emuluje port szeregowy. Umożliwia to transmisję danych użytkowych oraz danych konfiguracji wirtualnego portu szeregowego, tj. szybkości transmisji, ilości bitów stopu i danych w ramce, konfigurację parzystości i trybu sprzętowej kontroli przesyłania danych. Zastosowany mikrokontroler nie ma sprzężonego portu szeregowego z wirtualnym, dlatego możliwe było wykorzystanie danych konfiguracji portu szeregowego do komunikacji sterownika LED z komputerem PC. W skonstruowanym sterowniku LED konfiguracja

szybkości transmisji została zastosowana do zmiany trybu pracy. Aby komputer PC wykrył sterownik wymagane jest zainstalowanie programów obsługi urządzenia dostarczonych przez firmę Atmel, są one dostępne na stronie internetowej [12]. Skomunikowanie ze sterownikiem LED wymaga:

- połączenia komputera ze sterownikiem LED, co uzyskujemy poprzez przewód USB wpięty do portu *native* na PCB mikrokontrolera,
- skonfigurowania odpowiedniej szybkości transmisji w celu wyboru trybu pracy sterownika LED, w innym wypadku zostanie użyty tryb zapisany w pamięci nieulotnej sterownika,
- otwarcia portu szeregowego przypisanego sterownikowi LED przez system operacyjny komputera.

W celu zmiany trybu pracy sterownika LED należy zamknąć port szeregowy po stronie PC, a następnie otworzyć go z inną prędkością transmisji.

Zbudowany sterownik posiada 13 trybów pracy, dla których zdefiniowano formaty pakietów przesyłanych danych. W celu uniknięcia przesyłania zbędnych danych, w zależności od trybu pracy, zmieniający jest rozmiar pakietu i jego zawartość. Więcej informacji na temat protokołu komunikacyjnego znajduje się w pracy [3].

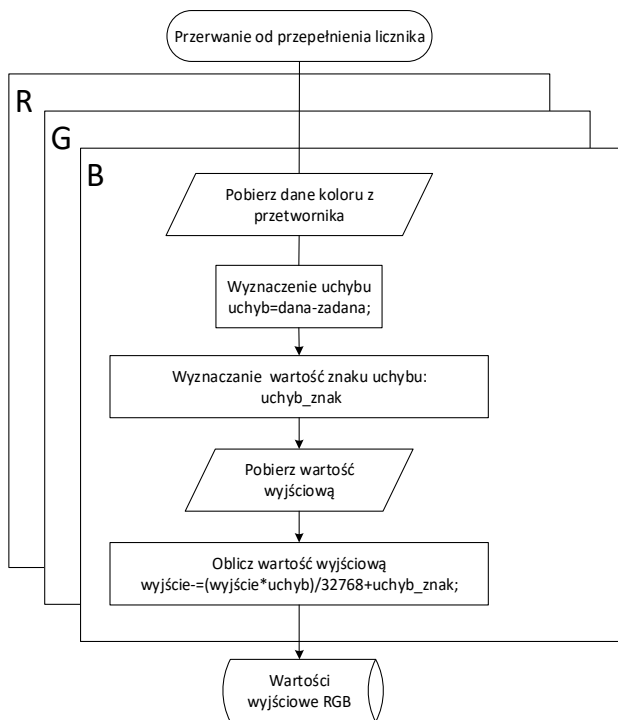
5. OPROGRAMOWANIE STEROWNIKA

Opracowany algorytm regulacji koloru i jasności diod LED spełnia następujące założenia:

- umożliwia regulację zarówno koloru jak i natężenia światła,
- jest wykonywany cyklicznie w odstępach czasu zbliżonych do czasu wykonywania pomiaru przez przetwornik koloru,
- zwraca liczby 16 bitowe, które są konwertowane do innych formatów w zależności od potrzeb, np. na format liczby 8 bitowej dla 8 bitowych kanałów DMX512,
- wartość zadana poszczególnych składowych kolorów RGB jest równoznaczna z wymaganym odczytem uzyskanym przez przetwornik koloru.

Założenie cykliczności rozwiązano poprzez zastosowanie, wbudowanego w mikrokontroler, licznika taktowanego zegarem MCU. Licznik ten po przepełnieniu uruchamia procedurę przerywania i zaczyna pracę od początku. Dzięki dobraniu odpowiedniego dzielnika zegara licznika uzyskano czas cyklu zbliżony do czasu pomiaru przetwornika koloru równego 100 ms. Spełnienie pozostałych założeń uzyskano dzięki zdefiniowaniu zmiennych oraz wykonaniu obliczeń.

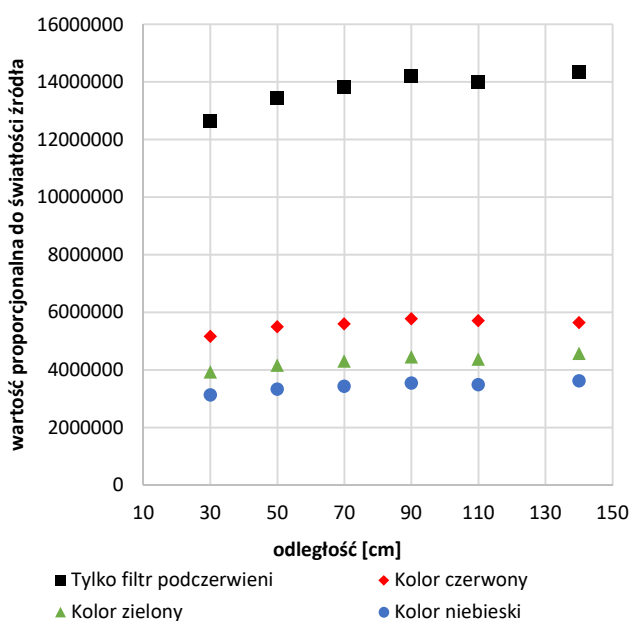
Algorytm zaimplementowany w pamięci mikrokontrolera oraz w aplikacji komputerowej przedstawiono na rysunku 3. Program dla mikrokontrolera został napisany w wersji 7 środowiska Atmel Studio, natomiast aplikacje komputerowe w wersji 2.1.2 środowiska Processing. Opracowana aplikacja komputerowa do sterowania manualnego i wykonywania pomiarów umożliwia manualne wpisywanie wartości określającej wypełnienia kanałów PWM sprzężonych z diodami LED oraz odczytywanie danych przychodzących z przetwornika koloru poprzez USB. Natomiast druga opracowana aplikacja umożliwia sterowanie automatyczne diodami LED poprzez USB. Diody LED RGB sterowane są w taki sposób, aby wartość światła poszczególnych składowych RGB padająca na przetwornik koloru odpowiadała wpisanym wartościom zadany.



Rys. 3. Zaimplementowany algorytm regulacji

6. POMIARY

W celu sprawdzenia liniowości przetwornika koloru wykonano serię pomiarów źródła światła z różnych odległości. Przetwornik koloru został ustawiony naprzeciwko świecącej diody LED, natomiast źródło światła i przetwornik koloru ustawiono równolegle. Dane z przetwornika koloru były odczytywane poprzez opracowaną aplikację. Uzyskane wyniki przedstawiono na rysunku 4 oraz tabeli 2, oś pionowa przedstawia wartość proporcjonalną do światłości źródła obliczoną na podstawie odczytów z przetwornika. Można zaobserwować tendencję do wzrostu wartości światłości przy większej odległości, tj. mniejszej ilości światła docierającej do przetwornika.

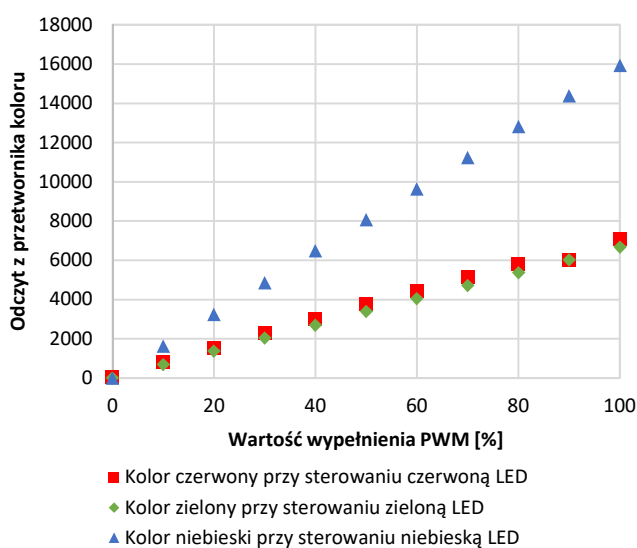


Rys. 4. Pomiar liniowości przetwornika koloru

Tabela 2. Wyniki pomiarów liniowości przetwornika koloru

Odległość przetwornika od źródła [cm]	3	5	7	9	11	14
Pomiar światła (tylko filtrem podczerwieni)	14050	5377	2820	1753	1158	732
Pomiar światła czerwonego	5745	2200	1143	714	472	288
Pomiar światła zielonego	4371	1670	881	550	362	234
Pomiar światła niebieskiego	3485	1333	701	438	289	185

Sprawdzono również liniowość wartości uzyskiwanego światła z diod LED w zależności od wartości wypełnienia PWM, wyniki pomiarów przedstawiono na rysunku 5 oraz w tabeli 3.



Rys. 5. Pomiar liniowości PWM

Tabela 3. Wyniki pomiarów liniowości PWM

Wypełnienie PWM		Odczyt wartości światła z przetwornika koloru odpowiadający kolorowi sterowanej diody LED		
0 – 4096	%	czerwony	zielony	niebieski
0	0	16	12	11
410	10	795	696	1639
819	20	1545	1375	3252
1229	30	2286	2050	4860
1638	40	3015	2700	6500
2048	50	3740	3395	8075
2458	60	4445	4050	9650
2867	70	5145	4720	11250
3277	80	5800	5375	12830
3686	90	6000	6030	14400
4096	100	7090	6675	15950

W celu wyznaczenia wzorców kolorów podstawowych, mieszanych i barwy białej o temperaturze światła dziennego przeprowadzono serię pomiarów przetwornikiem koloru i cyfrowym aparatem fotograficznym *Canon PowerShot A630*. W celu uzyskania pożądanej barwy ustawiono ręcznie, przy pomocy opracowanego programu, wartości wypełnienia sygnałów sterujących diodami elektroluminescencyjnymi do czasu uzyskania satysfakcjonującej barwy pomierzonej aparatem cyfrowym. Barwę z aparatu cyfrowego odczytywano,

jako wartości HSV (ang. *H* – Hue, *S* – Saturation, *V* – Value) [13]. Jako satysfakcjonujący poziom błędów przyjęto uchyb wartości odcienia kolorów poniżej 2°. Jednocześnie usiłowano uzyskać jak najwyższą wartość nasycenia w przypadku kolorów oraz jak najniższą wartość nasycenia dla bieli.

Na potrzeby implementacji wzorców wykonano pomiary dla różnych jasności poszczególnych kolorów i bieli oraz znormalizowano je do wartości odczytanych z przetwornika koloru bez filtra koloru. Uzyskane wzorce kolorów zostały wykorzystane jako punkty odniesienia w paletcie kolorów. Przy ich pomocy interpolowano wzorec koloru o pożądanym odcieniu i nasyceniu. W celu zmiany natężenia światła na pożądaną wszystkie wartości kolorów składowych uzyskanego wzorca są mnożone przez nieujemny współczynnik określający jasność. Współczynnik ten jest w przybliżeniu równy sumie odczytów poszczególnych kolorów składowych z przetwornika koloru po zakończonym procesie regulacji. Uzyskane wzorce dla zastosowanego przetwornika przedstawiono w tabeli 4.

Tabela 4. Uzyskane wzorce kolorów

Kolor	Czerwonony	Żółty	Zielony	Turkusowy	Niebieski	Fuksja	Biały
Składowa							
czerwona	0,872	0,407	0,073	0,057	0,020	0,518	0,306
zielona	0,044	0,427	0,605	0,484	0,152	0,104	0,373
niebieska	0,085	0,201	0,267	0,468	0,783	0,418	0,348

Aby wyznaczyć wartości zadane dla regulatora należy odpowiednio przeliczyć kolor i światłość (sumę składowych RGB odczytywanych przez przetwornik). Przykładowo wyznaczono dane wejściowe algorytmu regulacji LED RGB dla odcienia złożonego w 60% z koloru niebieskiego o nasyceniu 80% oraz dopełnionego turkusowym i współczynnika określającego światłość równego 2000. Uzyskanie niezbędnych danych wymagało:

1. Wyznaczenia wzorca dla pożądanego odcienia, złożonego w 60% z koloru niebieskiego dopełnionego turkusowym (tzn. kolorem z sąsiadującego narożnika), z istniejących wzorców kolorów na podstawie zależności

$$H_p = k_1 \cdot H_1 + (1 - k_1) \cdot H_2 \quad (1)$$

gdzie: H_p to wzorec dla pożądanego odcienia, H_1 - wzorec koloru najbliższego do pożądanego odcienia odczytany z tabeli 4, H_2 - wzorec koloru dopełniającego pożądanego odcień odczytany z tabeli 4, k_1 - wartość współczynnika określającego odcień.

Po podstawieniu odpowiednich danych otrzymujemy

$$H_p = 0,6 \cdot \begin{bmatrix} 0,020 \\ 0,152 \\ 0,783 \end{bmatrix} + (1 - 0,6) \cdot \begin{bmatrix} 0,057 \\ 0,484 \\ 0,468 \end{bmatrix} = \begin{bmatrix} 0,0348 \\ 0,2848 \\ 0,6570 \end{bmatrix}$$

2. Wyznaczenia wzorca, uwzględniającego dodatkowo pożądaną nasycenie równe 80%, na podstawie wzorca koloru białego i wzorca pożądanego odcienia na podstawie wzoru

$$H_{ps} = k_2 \cdot H_p + (1 - k_2) \cdot W \quad (2)$$

gdzie: H_{ps} to wzorec pożądanego odcienia o pożądanym nasyceniu, H_p - wzorec dla pożądanego odcienia, W - wzorec bieli odczytany z tabeli 4, k_2 - wartość nasycenia.

Po podstawieniu odpowiednich danych otrzymujemy

$$H_{ps} = 0,8 \cdot \begin{bmatrix} 0,0348 \\ 0,2848 \\ 0,6570 \end{bmatrix} + (1 - 0,8) \cdot \begin{bmatrix} 0,306 \\ 0,373 \\ 0,348 \end{bmatrix} = \begin{bmatrix} 0,0890 \\ 0,3024 \\ 0,5952 \end{bmatrix}$$

3. Wyskalowania uzyskanego wzorca pożądanego odcienia uwzględniającego wymagane nasycenie przez współczynnik określający jasność równy 2000, czyli

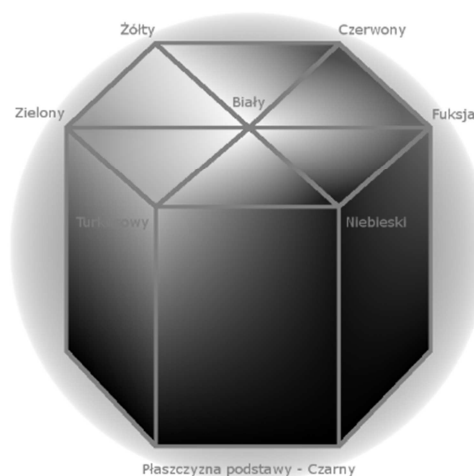
$$H_{psk} = k_3 \cdot H_{ps} \quad (3)$$

gdzie: H_{psk} to szukane nastawy dla algorytmu regulacji LED RGB, k_3 to współczynnik skalujący jasność.

Po podstawieniu danych otrzymujemy

$$H_{psk} = 2000 \cdot \begin{bmatrix} 0,0890 \\ 0,3024 \\ 0,5952 \end{bmatrix} = \begin{bmatrix} 178 \\ 605 \\ 1190 \end{bmatrix}$$

Wielościan kolorów (rys. 6) przedstawia rozmieszczenie kolorów względem siebie. Punkty znajdujące się na wierzchołkach górnego sześciokąta pokrywają się z uzyskanymi wzorcami kolorów, natomiast punkt środkowy pokrywa się z wzorcem bieli. Ponadto kolor czerni uzyskujemy dla zerowych wartości wejściowych algorytmu regulacji LED RGB. Natomiast maksymalną wartość jasności uzyskamy, gdy dowolny kanał LED ulegnie nasyceniu, tzn. wartość wypełnienia PWM po zakończonym procesie regulacji będzie równa 100%.



Rys. 6. Wielościan kolorów [3]

Na rysunku 7 przedstawiono wykonany sterownik LED RGB podczas pracy w trybie *MODE_PWM_S*, co oznacza, że algorytm regulacji wykonywany jest w mikrokontrolerze,

a sterowanie odbywa się przy pomocy przetwornika koloru oraz generatora PWM. Światło emitowane przez diody ma odcień niebiesko - turkusowy, zgodnie z wartościami zadanymi znajdującymi się w pamięci nieulotnej mikrokontrolera.



Rys. 7. Sterownik LED w czasie pracy

7. ZAKOŃCZENIE

Zbudowane urządzenie może być wykorzystane jako narzędzie do pomiaru natężenia i koloru oświetlenia. Wykonywanie pomiarów światła przetwornikiem koloru można wykorzystać w pracowniach graficznych w celu stabilizacji temperatury światła wewnątrz pomieszczeń. Można je ponadto wykorzystać jako stanowisko laboratoryjne do obserwacji zachowania diod LED, a w szczególności zagadnień związanych z uzyskaniem barwy białej. Możliwe jest rów-

nież wykorzystanie układu jako kontrolera protokołu DMX512 do sterowania efektami scenicznymi, wymaga to jednak zastosowania odpowiednich buforów, np. układu scalonego MAX485 firmy Maxim. Zbudowany sterownik LED, po modyfikacji programu mikrokontrolera, można rozbudować o kolejne przetworniki koloru i moduły LED w celu sterowania oświetleniem w wielu pomieszczeniach. Program mikrokontrolera można też przystosować do pracy z innym przetwornikiem koloru, np. z wykorzystującym filtry interferencyjne w celu zwiększenia dokładności odczytów światła.

8. BIBLIOGRAFIA

1. <https://www.adafruit.com/datasheets/TCS34725.pdf>
2. <http://www.mazet.de/de/dokumente/data-sheets/mcscif/download>
3. Pęczkowski D.: Projekt i realizacja sterownika diod LED RGB mocy, praca dyplomowa inżynierska, WE, KAO, AM Gdynia, 2016
4. http://www.qlt.it/public/allegati/08_RGBLEDSYSTEM_S187-216_15.pdf
5. <https://pl.wikipedia.org/wiki/DMX512>
6. <http://www.digikey.com/en/product-highlight>
7. <http://www.arduino.org/products>
8. <https://learn.adafruit.com/adafruit-color-sensors/>
9. <https://people.xiph.org/~xiphmont/thinkpad/pt4115e.pdf>
10. <https://www.adafruit.com/datasheets/PCA9685.pdf>
11. <https://learn.adafruit.com/adafruit-16-channel-pwm-slash-servo-shield/overview>
12. <https://gallery.atmel.com/Products/Details/>
13. https://pl.wikipedia.org/wiki/HSV_grafika

AUTOMATED HIGH POWER RGB LED DRIVER

Automation system that controls electro luminescent diodes which emit the light of basic colors, red, green and blue, was presented in this article. Brightness and color of the light corresponds to the parameters set in the computer application created to control the driver.

Key words: the measurement of colors, RGB LED driver, microcontroller.