



# Regulator natężenia oświetlenia z Wi-Fi

Na łamach *Elektroniki Praktycznej* prezentowano różnorodne regulatory oświetlenia popularnie zwane „ściemniaczami”. O ile zasada działania opisywanego urządzenia jest taka sama, jak poprzedników i polega na zmianie czasu włączenia triaka, to interfejs użytkownika jest bardzo nowoczesny, ponieważ wykonano go z użyciem smartfonu i sieci Wi-Fi. Sprawia to, że za pomocą jednej aplikacji jesteśmy w stanie regulować oświetlenie nawet w całym domu. Dodatkowo, układ umożliwia sterowanie dwoma niezależnymi kanałami, co podnosi walory użytkowe i umożliwia tworzenie scen świetlnych w pokoju.

**Rekomendacje:** regulator może przydać się w pokoju dziecięcym lub w pomieszczeniu ze sprzętem audio-wideo.

Zasada działania regulatora polega na zmianie czasu trwania połówek sinusoidy i przypomina modulację PWM, co pokazano na **rysunku 1**. Elementem regulacyjnym jest triak, którego moment załączenia jest wyznaczany i zmieniany przez mikrokontroler.

Schemat ideowy układu regulatora pokazano na **rysunku 2**. Układ ściemniacza podzielono na cztery bloki funkcjonalne: zasilacz z detektorem przejścia fazy przez zero, mikrokontroler, elementy wykonawcze mocy oraz moduł komunikacji Wi-Fi.

Zasilanie urządzenia wykonano w oparciu o typowy, małogabarytowy transformator sieciowy. Zrobiono to z trzech powodów:

Galwaniczne odizolowanie części niskonapięciowej układu od napięcia sieci. Ma to szczególne znaczenie, ponieważ płytka drukowana zaprojektowano dla komponentów SMD. Aby po zmontowaniu układ działał prawidłowo, należy zaprogramować mikrokontroler. Do tego przewidziano pady na płytce do wlotowania przewodów połączeniowych programatora. Jest to czynność

## DODATKOWE MATERIAŁY NA FTP:

<ftp://ep.com.pl>

USER: 07643, PASS: 332wwppm

## W ofercie AVT\*

**AVT-5530 A, B, UK**

### Podstawowe informacje:

- Dwustronna, okrągła płytka drukowana o średnicy 55 mm.
- Możliwe zamontowanie w typowej puszcze instalacyjnej.
- Obciążenie: żarówka o mocy do 60 W lub inne źródło światła kompatybilne z regulatorami fazowymi.
- Połączenie z systemem nadrzędnym za pomocą Wi-Fi.
- Podstawowa aplikacja wykonana dla systemu Android (kontrolowanie oświetlenia za pomocą smartfonu lub tabletu).
- Mikrokontroler ATmega8, moduł Wi-Fi dystrybuowany przez firmę Atmel.

### Projekty pokrewne na FTP:

(wymienione artykuły są w całości dostępne na FTP)

- AVT-1869 Wzmacniacz do taśm RGB (EP 8/2015)
- AVT-5504 Kontroler oświetlenia RGB z Bluetooth (EP 6/2015)
- AVT-3133 Sterownik oświetlenia LED sterowany dowolnym pilotem (EdW 4/2015)
- AVT-1847 Miniatury sterownik taśmy LED (EP 2/2015)
- AVT-5487 PWMLEDz: 10-kanałowy sterownik taśm LED z interfejsem Modbus lub SPPoB (EP 1/2015)
- AVT-1800 LED Dimmer – regulator oświetlenia LED (EP 5/2014)
- AVT-5400 DMX Dimmer & Relay – regulator oświetlenia i wyłącznik z interfejsem DMX (EP 6/2013)
- AVT-5376 RadioDimmer – regulator oświetlenia w mieszkaniu (EP 1/2013)
- AVT-5361 4dimmer – 4-kanałowy regulator oświetlenia (EP 9/2012)
- AVT-5295 Blue Relay – moduł przekaźników sterowany przez Bluetooth (EP 6/2011)
- AVT-1545 Programowany sterownik światła (EP 10/2009)
- AVT-1509 Sterownik RGB (EP 2/2009)
- AVT-2890 Sterownik Bluetooth (EdW 1/2009)
- AVT-5164 RGB Driver (EP 12/2008)
- AVT-2794 Automatyk sterownik oświetlenia (EdW 8/2006)
- AVT-2749 4-kanałowy regulator oświetlenia (EdW 3/2005)
- AVT-2210 Najprostszy regulator mocy 230 V (EdW 3/1997)

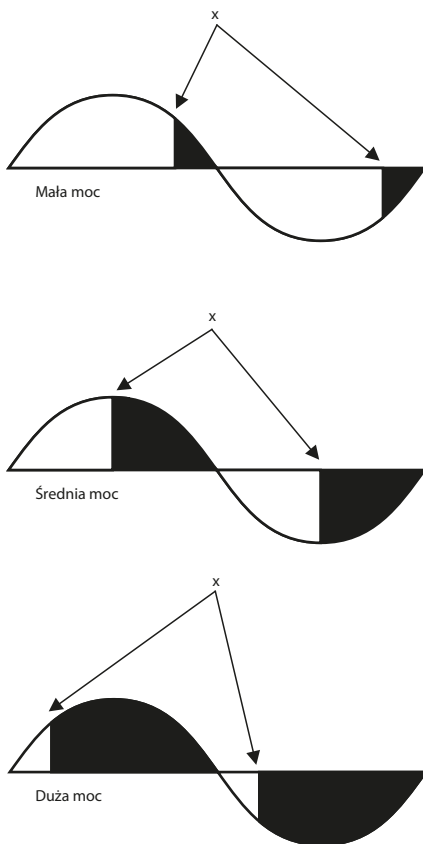
\* Uwaga: Zestawy AVT mogą występować w następujących wersjach:  
 AVT xxxx UK to zaprogramowany układ, tylko i wyłącznie. Bez elementów dodatkowych.  
 AVT xxxx A płytka drukowana PCB (lub płytki drukowane, jeśli w opisie wyraźnie zaznaczono), bez elementów dodatkowych.  
 AVT xxxx A+ płytka drukowana i zaprogramowany układ (czyli połączenie wersji A i wersji UK) bez elementów dodatkowych.  
 AVT xxxx B płytka drukowana (lub płytki) oraz komplet elementów wymienionych w załączniku pdf  
 AVT xxxx C to nic innego jak zmontowany zestaw B, czyli elementy wlotowane w PCB. Należy mieć na uwadze, że o ile nie zaznaczono wyraźnie w opisie, zestaw ten nie ma obudowy ani elementów dodatkowych, które nie zostały wymienione w załączniku pdf oprogramowanie (nieczęsto spotykana wersja, lecz jeśli występuje, to niezbędne oprogramowanie można ściągnąć, klikając w link umieszczony w opisie kitu)  
 AVT xxxx CD oprogramowanie (nieczęsto spotykana wersja, lecz jeśli występuje, to niezbędne oprogramowanie można ściągnąć, klikając w link umieszczony w opisie kitu)  
 Nie każdy zestaw AVT występuje we wszystkich wersjach! Każda wersja ma załączony ten sam plik pdf! Podczas składania zamówienia upewnij się, którą wersję zamawiasz! (UK, A, A+, B lub C). <http://sklep.avt.pl>

jednorazowa. Gdyby nie było izolacji galwanicznej, to takie programowanie groziłoby porażeniem prądem elektrycznym.

Wymagana wydajność prądowa do zasilania modułu Wi-Fi. Podczas pracy chwilowy pobór prądu przez moduł może wynosić nawet 250 mA, co trudno byłoby uzyskać stosując rozwiązanie z zasilaczem beztransformatorowym z kondensatorem pełniącym rolę „rezystora” obniżającego napięcie.

Chęć wykonania układu detekcji przejścia fazy przez zero bez zbędnego trzcenia

x – punkt załączenia triaka



Rysunek 1. Zasada regulacji fazowej oparta jest o sterowanie czasem otwarcia triaka w punkcie x. Im szybciej to nastąpi po przejściu napięcia sieci zasilającej przez zero, tym więcej energii dostanie odbiornik i tym jaśniej świeci żarówka

**Wykaz elementów**

**DAC1955 – część analogowa**

**Rezystory:** (SMD 0805)

- R1, R4: 10 kΩ
- R2: 1 kΩ
- R3: 4,7 kΩ
- R5, R7: 220 Ω
- R6, R8: 560 Ω

**Kondensatory:** (SMD 0805)

- C1, C2, C6: 100 nF
- C5, C8: 22 pF
- C7: 10 nF
- C3: 470 μF/10 V (przewlekany, niski profil)
- C4: 220 μF/6,3 V (przewlekany, niski profil)

**Półprzewodniki:**

- U1: LM1117-3.3 (SOT223)
- U2: ATmega8 (TQFP32)
- B1: mostek prostowniczy SMD
- D1: dioda prostownicza (obudowa SMA)
- T1: BC847
- OK1, OK2: MOC3021 (SMD)
- TR1, TR2: Z0103MN (SOT223)

**Inne:**

- Transformator sieciowy zalewany TEZ1.5/ D/6V
- Q1: rezonator kwarcowy 11.0592 MHz, niski profil
- Mod\_WiFi: Moduł ATNEL-WIFI-232
- X1, X2, X3: złącza ARK2

```
Listing 1. Analizowanie transmisji szeregowej
void analizuj_dane_rs232 (char *buf);
register_uart_str_rx_event_callback(analizuj_dane_rs232 ); // rejestrowanie
funkcji do analizowania danych odebranych przez UART

while(1) // pętla główna programu
{
    UART_RX_STR_EVENT(uart_buf);
} // while

void analizuj_dane_rs232 (char *buf)
{
    char *wsk;
    wsk=strtok(buf,"?");
    if(!strcmp(wsk,"out1"))
    {
        wsk=strtok(NULL,"?"); // w zmiennej wsk jest teraz wartość
zadana out1
        pwm1=atoi(wsk);
    }
    if(!strcmp(wsk,"out2"))
    {
        wsk=strtok(NULL,"?"); // w zmiennej wsk jest teraz wartość
zadana out2
        pwm2=atoi(wsk);
    }
    if(!strcmp(wsk,"AT+RST")) // **** reset programu do aktualizacji
oprogramowania
    {
        cli(); // disable interrupts
        wdt_enable( 0 ); // set watchdog
        while(1); // wait for RESET
    }
}
```

energii elektrycznej. Najprościej byłoby zastosować transformator z rezystorem ograniczającym jego prąd, ale taki układ ma jedną wadę. Otóż energia wytracana w rezystorze zamienia się w ciepło. Dodatkowo, powoduje niepotrzebne zużywanie energii elektrycznej, co w skali roku może dać nawet kilka kilowatogodzin. Zakładając pracę takiego regulatora przez kilka lat, jest to nie do przyjęcia. Dlatego też zastosowano układ przejścia przez zero w dość nietypowym rozwiązaniu. Za mostkiem prostowniczym zastosowano diodę prostowniczą w roli separatora. Transzystor T1 pracuje w roli klucza – przez większą część okresu jest zamknięty, otwiera się tylko w momencie przejścia napięcia sieci przez zero.

Pozostałe elementy za diodą D1 tworzą typową aplikację stabilizatora napięcia 3,3 V. Na uwagę zasługuje tylko kondensator C3, który powinien mieć pojemność co najmniej niż 470 μF.

Pracę regulatora nadzoruje mikrokontroler ATmega8. Jego zadaniem jest odbieranie danych z modułu Wi-Fi, wykrywanie momentu przejścia sieci zasilającej przez zero i odpowiednie sterowanie triakami. Jest to jego podstawowa aplikacja, która raczej nie potrzebuje omawiania. Komunikacja z modułem Wi-Fi jest oparta o transmisję UART z prędkością 115,2 kb/s.

Do wykrywania przejścia fazy przez zero użyto przerwania zewnętrznego INT0. W jego obsłudze jest zerowany licznik Timer2, który to z kolei odpowiada za odmierzenie odcinków 50 μs. Pozwala to na zmianę czasu włączenia triaka w 200 krokach, co odpowiada regulacji od 0% do 100% mocy wyjściowej.

**Oprogramowanie**

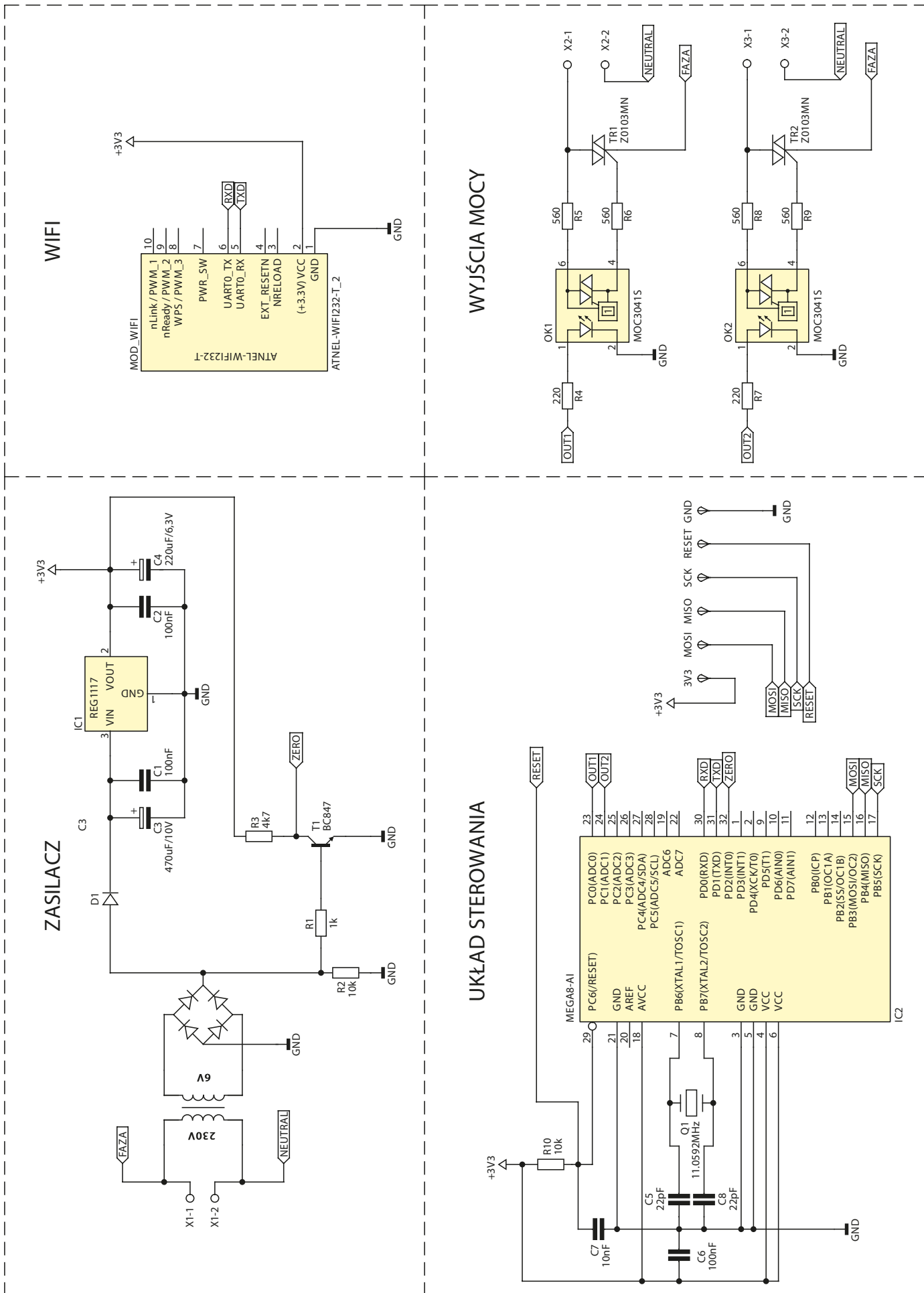
Program mikrokontrolera napisano w całości w języku C. Jego działanie sprowadza się

do wykrycia przejścia przez zero fazy sieci zasilającej i wysterowaniu triaka w odpowiedniej chwili – zależnej od żądanej jasności świecenia żarówki. Mikrokontroler wykonuje te czynności obsługując przerwanie.

W pętli głównej mikrokontroler przez cały czas sprawdza czy odebrano dane z aplikacji sterującej za pomocą UART. Jeśli tak, to zapisuje je do bufora odbiorczego i poddaje analizie. Ponieważ dane przesyłane są w postaci znaków ASCII (komenda) i dane przedzielone separatorami, więc zastosowano mechanizm tokenów i parsowania danych. Oparty go o tzw. wywołania *callback*. Może brzmieć to groźnie, ale to tylko pozory. Po szczegóły odsyłam do książki „Mikrokontrolery AVR, Język C, podstawy programowania” autorstwa Mirosława Kardasia. Jest tam bardzo przystępnie i dokładnie wyjaśniony cały mechanizm z przykładami.

Wybrane fragmenty programu związane z analizowaniem transmisji szeregowej pokazano na **listingu 1**. W pętli głównej program oczekuje na dane przesyłane z aplikacji, które mają postać: out1?25 lub out2?55, co po analizie oznacza – kanał OUT1 25% mocy, kanał OUT2 55% mocy. Separatorami jest znak zapytania?

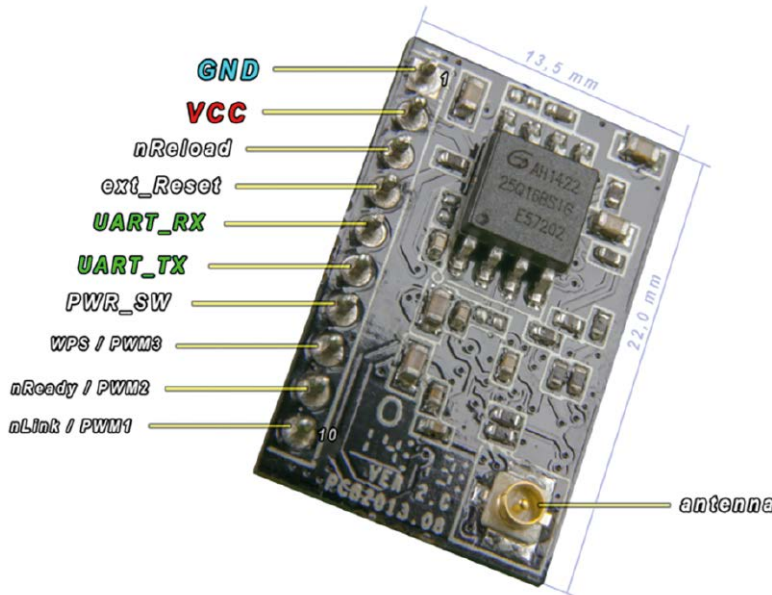
Spostrzegawczy Czytelnicy zapewne zauważyli również komendę odpowiedzialną za uruchomienie watchdoga i zerowanie mikrokontrolera – AT+RST. Umożliwia ona zdalny restart, a co za tym idzie – zdalną zmianę oprogramowania mikroprocesora ATmega8. I tu ujawnia się kolejny atut urządzenia – jego oprogramowanie można zaktualizować zdalnie, bez konieczności demontażu procesora, przyłączania programatora i wszystkimi uciążliwościami z tym związanymi. Pomocny będzie program **MKBootloader**, który jest to aplikacją



Rysunek 2. Schemat ideowy bezprzewodowego regulatora natężenia oświetlenia



# ATNEL-WIFI232-T



Rysunek 3. Moduł ATNEL-WIFI-232

umożliwiająca komunikację z mikrokontrolerem w celu aktualizacji zawartości pamięci Flash za pomocą trzech różnych interfejsów: RS232, Bluetooth oraz WiFi. Program jest dostępny na stronie internetowej firmy Atmel.

Triaki TR1 i TR2 są elementami wykonawczymi. W obwodach ich bramek zastosowano izolację galwaniczną uzyskaną za pomocą optotriaków typu MOC3021. Jest to typowa aplikacja i nie wymaga szczególnego omawiania.

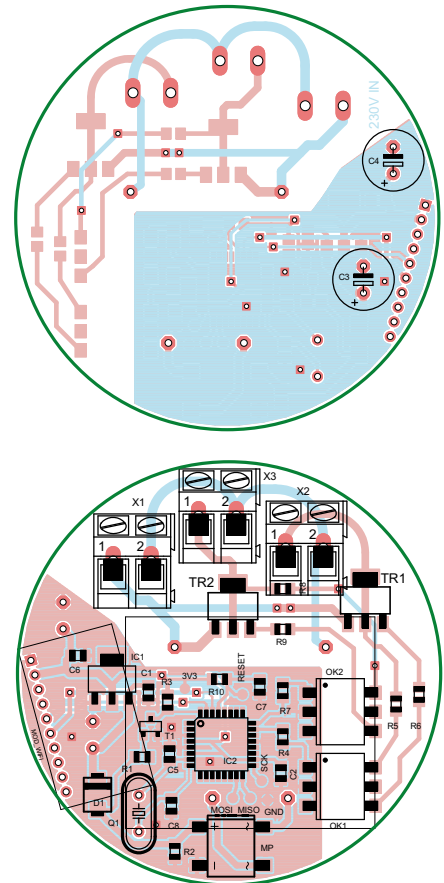
## Moduł Wi-Fi

Na rynku jest dostępnych tak dużo różnych modułów WiFi, że wybór jest naprawdę trudny. Dla większości z nich dostępna jest dokumentacja w języku angielskim lub chińskim i ktoś nieznający języka może mieć trudność w aplikacji modułu. W wypadku napotkania problemów niezmiernie ważny jest właściwy poziom wsparcia technicznego ze strony producenta lub dystrybutora. Mając te aspekty na względzie zdecydowałem się na zastosowanie pokazanego na **rysunku 3** modułu ATNEL-WIFI232-T dystrybuowanego przez firmę Atmel ze Szczecina. Podstawowe parametry techniczne modułu ATNEL WIFI232-T są następujące:

- Wyprowadzenia w rastrze 2 mm (10 pinów).
- Zasilanie +3,3 V/do 250 mA.
- Wyprowadzenia GPIO/PWM.
- Wyprowadzenia RxD i TxD interfejsu UART.
- Możliwość konfigurowania i sterowania za pomocą komend AT.
- Gniazdo dla anteny zewnętrznej 2,4 GHz.
- Częstotliwość pracy 2,4 GHz.
- Obsługa standardów Wi-Fi b/g/n.
- Wsparcie dla funkcji WPS.

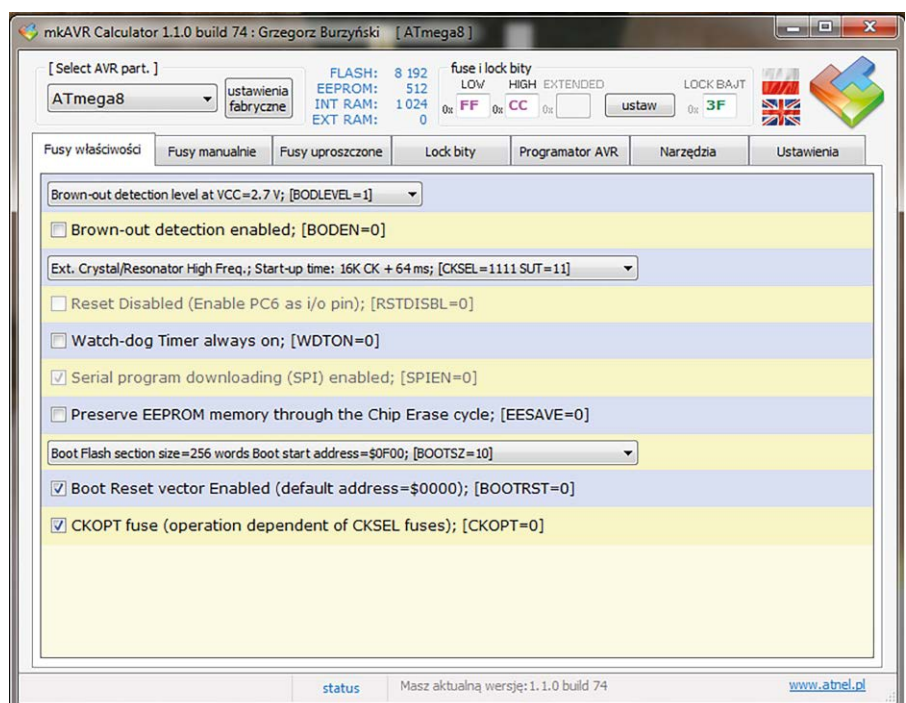
- Obsługa zabezpieczeń WEP, WPA, WPA2 oraz szyfrowania AES.
- Wyprowadzenie umożliwiające zarządzanie poborem prądu.
- Małe wymiary: 22 mm×13,5 mm×6 mm.
- Certyfikat FCC/CE.

Jest to **moduł Wi-Fi małej mocy** przeznaczony do pracy w systemach wbudowanych. Podstawową zaletą modułu jest łatwość opanowania protokołu komunikacyjnego, którą udało się uzyskać dzięki innowacyjnemu pomysłowi na wykorzystanie tzw. trybu transparentnego. W odróżnieniu od szeregu produktów konkurencyjnych, komunikację wykorzystującą protokołu TCP/UDP, ale także http i inne można sprowadzić do jednego

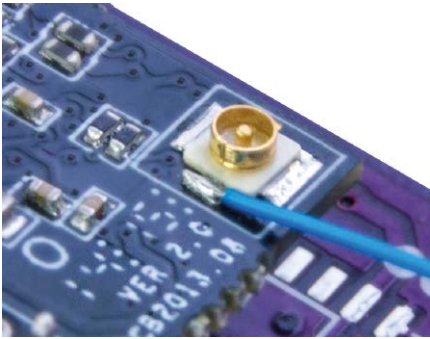


Rysunek 4. Schemat montażowy bezprzewodowego regulatora natężenia oświetlenia

– nieskomplikowanej komunikacji za pomocą UART. Kolejne cechy stawiające moduł w pierwszym rzędzie wraz z najlepszymi, to sprzętowe wsparcie nie tylko dla takich trybów pracy, jak STA czy AP, lecz także dla trybu mieszanego AP+STA. W każdej konfiguracji moduł umożliwia dostęp do podstawowych



Rysunek 5. Ustawienie bitów konfiguracyjnych mikrokontrolera ATmega8



**Fotografia 6. Prawidłowy przyłutowany 3-centymetrowy odcinek przewodu przy gnieździe antenowym**

funkcji konfiguracyjnych za pomocą wbudowanej strony www i to strony w języku polskim. Dodatkowo, moduł jest zaopatrzony w autorski firmware, który umożliwia np. pobieranie dokładnego czasu z NTP z dowolnej strefy czasowej, wybudzenie zewnętrznych urządzeń, np. komputerów PC za pomocą funkcji WOL (Wake On LAN) czy sprawdzanie adresu IP zdalnego klienta, który ostatnio połączył się z modulem (tylko w trybie TCP). Dodatkowo, firma Atmel udostępnia specjalistyczne narzędzie konfiguracyjne w postaci programu *ATB WiFi Config*, dzięki któremu nie musimy męczyć się z konfigurowaniem modułu za pomocą komend AT. Każdy, kto choć raz miał okazję używać komend AT przy konfigurowaniu np. modułów GSM wie, jakie jest to uciążliwe. Dodatkowo, każdy klient może liczyć na wsparcie techniczne ze strony firmy i to praktycznie 24 godziny na dobę. Wystarczy tylko wejść na forum firmowe <http://forum.atmel.pl/portal.php>.

## Montaż i uruchomienie

Schemat montażowy regulatora oświetlenia pokazano na **rysunku 4**. Dla regulatora zaprojektowano płytkę drukowaną w kształcie koła o średnicy 55 mm. Ma to na celu jej łatwe umieszczenie w puszcze przelącznika światła lub w puszcze na ścianie.

Montaż regulatora należy rozpocząć od przyłutowania wszystkich elementów SMD zaczynając od mikrokontrolera. Na koniec należy przyłutować złącza ARK, podstawkę pod moduł Wi-Fi i transformator sieciowy. Przed włożeniem modułu Wi-Fi w podstawkę należy włączyć urządzenie i sprawdzić napięcie występujące pomiędzy wyprowadzeniami 1 i 2 złącza. Powinno ono wynosić 3,3 V. Jeśli tak jest, to odłączamy zasilanie sieciowe i do specjalnie wyprowadzonych padów na płytce lutujemy przewody programatora i ponownie włączamy zasilanie. Ta czynność jest zbędna, jeśli mamy wcześniej zaprogramowany mikrokontroler. Jeśli nie, to należy go zaprogramować.

Jako pierwsze musimy ustawić fuse bity. Osobiście korzystam z programu MKAVRCalculator, prawdziwego „kombajnu”, który ułatwia programowanie bitów konfiguracyjnych *fuse bit* oraz zabezpieczających *lock bit* w mikrokontrolerach AVR. Ma on też wiele innych ciekawych funkcji, których nie znajdziemy gdzie indziej. Musimy tak zaprogramować bity konfiguracyjne, aby umożliwić taktowanie ATmegi za pomocą rezonatora kwarcowego o częstotliwości 11,0592 MHz, ustawić rozmiar sekcji bootloadera na 256 słów [BOOTSZ=10] i zmienić adres wektora resetu, aby program startował z sekcji bootloadera. Odpowiednie ustawienia pokazano na **rysunku 5**.

Jeśli Fusebity już zaprogramowaliśmy, należy wgrać dołączony w materiałach dodatkowych program *dimmer.hex*. **Oprogramowanie współpracuje wyłącznie z modułami dystrybuowanymi przez firmę Atmel.** Te same moduły pochodzące od innych firm mogą nie działać prawidłowo. Ma to związek, jak wspomniano, ze zmienionym firmwarem modułów Atmel.

## Konfigurowanie modułu Wi-Fi

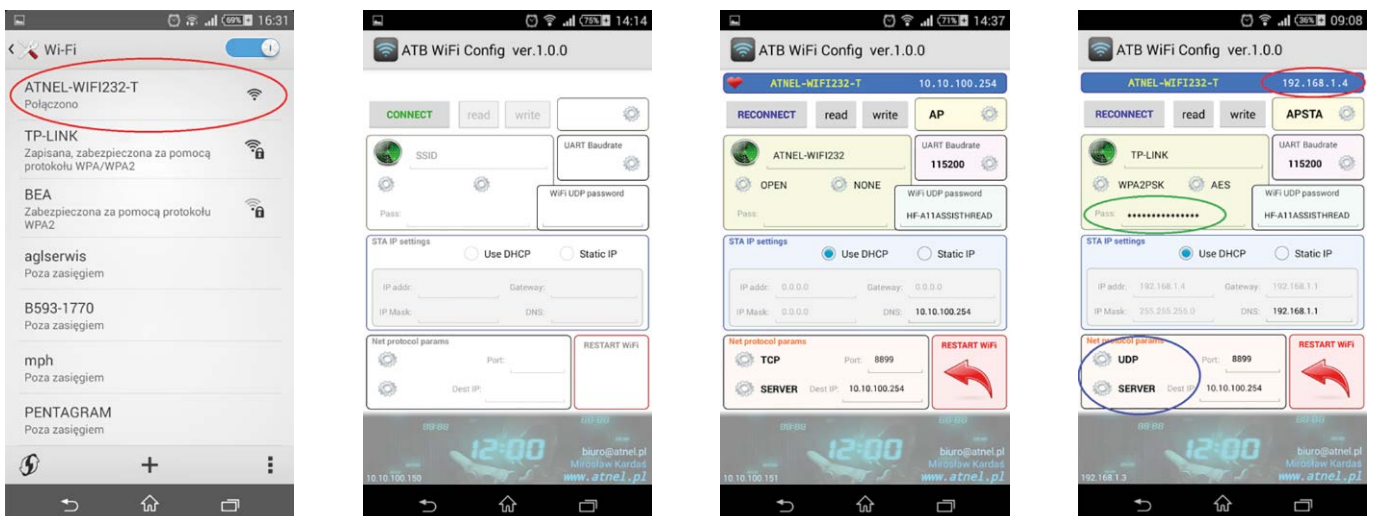
Po prawidłowym zaprogramowaniu mikrokontrolera wyłączamy zasilanie i wkładamy

w podstawkę moduł Wi-Fi. Moduł ma wprowadzone gniazdo do podłączenia anteny zewnętrznej. W jej roli należy, tak jak pokazano na **fotografii 6**, przyłutować 3-centymetrowy odcinek drutu w izolacji. Z taką anteną moduł pracuje poprawnie, a siła odbieranego sygnału jest niewiele niższa, niż przy zastosowaniu specjalizowanej anteny zewnętrznej.

Kolejnym krokiem jest skonfigurowanie modułu Wi-Fi. Wykonamy to za pomocą naszego telefonu. Aplikację **ATB WiFi Config** należy pobrać ze sklepu Google i zainstalować. Należy się również upewnić, że jesteśmy w zasięgu naszej domowej sieci Wi-Fi oraz znać podstawowe dane dotyczące ustawień routera.

Załączamy zasilanie regulatora. Ponieważ moduły WiFi domyślnie pracują w trybie AP, więc w pierwszej kolejności łączymy za pomocą telefonu z modulem, jak to pokazano na **rysunku 7a**. Następnie uruchamiamy program *ATBWIFConfig*, co spowoduje wyświetlenie ekranu głównego, jak na **rysunku 7b**. Pierwszą czynnością to odnalezienie modułu Wi-Fi. W tym celu wybieramy przycisk *CONNECT*, wyszukujemy moduł i łączymy się z nim, a następnie odczytujemy jego konfigurację za pomocą *READ*. Powinniśmy uzyskać odczyt taki, jak na **rysunku 7c**. Jak widać moduł pracuje w trybie AP. Zmieniamy tryb pracy na AP/STA, klikamy na *WRITE* i restartujemy moduł za pomocą *RESTART WIFI*. Po restarcie powinniśmy zobaczyć, że tryb pracy zmienił się na AP/STA. Jeśli tak jest, to następnie klikamy na symbol radaru i wyszukujemy domową sieć Wi-Fi. Po jej wybraniu moduł automatycznie rozpoznaje typ (**rysunek 7d**) zabezpieczeń. Pozostaje nam wpisanie hasła dostępu do sieci (zielona obwódka na rys. 7d). Ponownie zapisujemy konfigurację za pomocą *WRITE* i restartujemy moduł *RESTART WIFI*.

Wyłączamy program. Następnie przełączamy się w telefonie na połączenie Wi-Fi



**Rysunek 7. Oprogramowanie ATBWIFConfig: a) łączenie telefonu z modulem, b) ekran główny programu, c) konfiguracja odczytana z modułu, d) ustalenie typu zabezpieczeń**



Rysunek 8. Okno główne aplikacji regulatora oświetlenia

z naszą siecią domową i ponownie uruchamiamy program. Jeśli wszystko zrobiliśmy prawidłowo, to moduł powinien otrzymać adres IP z puli adresów przyznawanych przez nasz router (czerwona obwódka na rys. 7d). Ostatnią czynnością jest ustawienie w zakładce *NET Protocol Params* typu połączeń na *UDP SERVER*.

Ostatni raz wybieramy *WRITE* i *RESTART*. Moduł po restarcie jest gotowy

do pracy. W razie problemów ze skonfigurowaniem modułu pomocy można szukać na stronie internetowej firmy Atnel.

Aplikację na telefon – WHLD.apk (Wireless Home Light Dimmer – **rysunek 8**) napisano w środowisku *Basic4Android*. Przyznam, że nie spodziewałem się, że okaże się to tak łatwe. Ponownie bardzo pomocne okazały się filmiki opublikowane przez firmę Atnel i dostępne w serwisie youtube. Do wysyłania danych po UDP używamy tylko dwóch poleceń – *Packet.Initialize(data, ipaddress, 8899)* oraz *UDPSocket1.Send(Packet)*. Zmienna *data* to dane do wysłania, *ipaddress* to adres urządzenia docelowego, 8899 to numer portu. Dodam, że działający szkielet aplikacji powstał w przeciągu dosłownie kilku godzin, a nigdy wcześniej nie pisałem programów dla systemu Android.

Po uruchomieniu aplikacji naciskamy symbol kluczy w lewym dolnym rogu, a następnie symbol lupy, aby wyszukać moduł Wi-Fi. Jest to operacja jednorazowa, ponieważ aplikacja zapamiętuje wprowadzone dane. Na ekranie głównym mamy do dyspozycji dwa suwaki odpowiedzialne umożliwiające regulowanie oświetlenia w kanale 1 i kanale 2, w zakresie 0...100%. Każdorazowo po uruchomieniu aplikacja

odczytuje aktualną pozycję suwaków, więc nawet po wyłączeniu i ponownym włączeniu telefonu na ekranie zostaną przywrócone ostatnio ustawione wartości. Pod każdym suwakiem widoczne są również przyciski ON i OFF. Służą one do włączenia/wyłączenia oświetlenia, ale odbywa się to płynnie, co podnosi walory użytkowe całego urządzenia. Aplikacja jest do pobrania ze sklepu Google.

Na koniec chciałbym napisać, że w ramach doskonalenia pisania aplikacji na system Android powstaje kolejna wersja do obsługi dimmera. W dniu oddania artykułu do druku jest dostępne sterowanie tylko jednym modułem. W założeniach ma być dodana obsługa wielu modułów, aby za pomocą pojedynczej aplikacji można było sterować oświetleniem w całym domu. Zachęcam czytelników do własnych eksperymentów z programowaniem. Na stronie producenta B4A można pobrać wersję 30-dniową bez ograniczeń i samodzielnie spróbować napisać aplikację. Zapewniam, że jest to łatwe.

Chciałbym podziękować panu Mirosławowi Kardasiowi za wsparcie techniczne oraz poświęcony czas podczas pisania aplikacji regulatora oświetlenia.

**Grzegorz Burzyński**  
sp5ein@gmail.com