



W ofercie AVT\*

**AVT5943**

**Podstawowe parametry:**

- uzyskuje się pole widzenia o zakresie 180° zarówno w płaszczyźnie horyzontalnej, jak i wertykalnej dla kamerki ze standardowym obiektywem,
- sterowanie modułem kamery odbywa się za pośrednictwem przeglądarki internetowej i stron generowanych przez sam moduł,
- możliwość sterowania kamerą i ramieniem za pomocą rozkazów wpisywanych bezpośrednio w pasku przeglądarki i przesyłanych w trybie GET.

\* Uwaga! Elektroniczne zestawy do samodzielnego montażu. Wymagana umiejętność lutowania! Podstawową wersją zestawu jest wersja [B] nazywana potocznie KIT-em (z ang. zestaw). Zestaw w wersji [B] zawiera elementy elektroniczne (w tym [UK] – jeśli występuje w projekcie), które należy samodzielnie wylutować w dołączonej płytce drukowanej (PCB). Wykaz

elementów znajduje się w dokumentacji, która jest podlinkowana w opisie kitu. Mając na uwadze różne potrzeby naszych klientów, oferujemy dodatkowe wersje:
 

- wersja [C] – zmontowany, uruchomiony i przetestowany zestaw [B] (elementy wylutowane w płytce PCB)
- wersja [A] – płytka drukowana bez elementów i dokumentacji

**Dodatkowe materiały do pobrania ze strony [www.ulubionykiosk.pl/media](http://www.ulubionykiosk.pl/media)**

- AVT5897 Sterownik mikroserwomechanizmów do Pi Pico (EP 11/2021)
- AVT5886 16 kanałowy kontroler PWM – sterownik serwomechanizmów dla RPi Zero (EP 9/2021)
- AVT5752 Rozgątełnik dla serwomechanizmów (EP 3/2020)
- AVT5731 Radiowy sterownik serwomechanizmów (EP 12/2019)
- AVT1632 Tester serwomechanizmów modelarskich (EP 8/2011)
- AVT5290 3-kanałowa aparatura do zdalnego sterowania modelem (EP 5/2011)
- AVT1605 Dwustanowy sterownik serwomechanizmu (EP 2/2011)
- AVT1570 Spawalniacz serwomechanizmu (EP 5/2010)

Kity, w których występuje układ scalony wymagający zaprogramowania, mają następujące dodatkowe wersje:
 

- wersja [A+] – płytka drukowana [A] + zaprogramowany układ [UK] i dokumentacja
- wersja [UK] – zaprogramowany układ

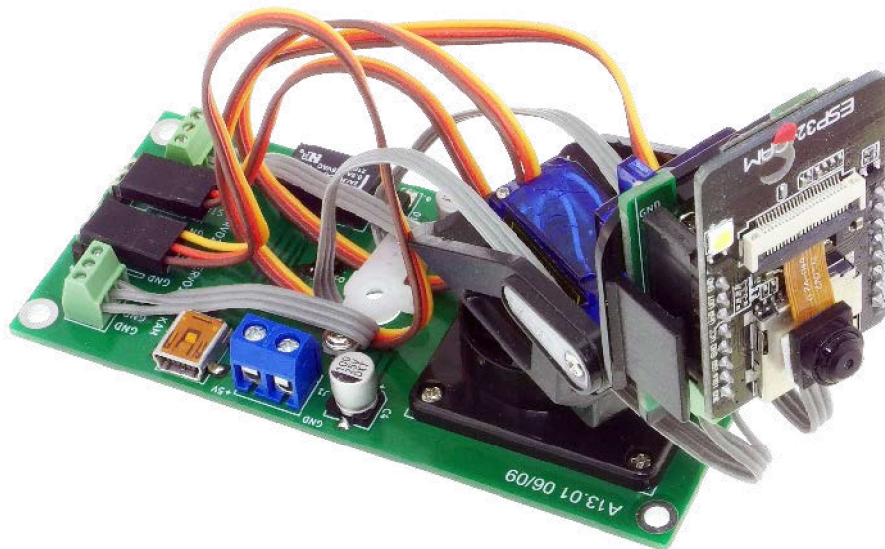
 Nie każdy zestaw AVT występuje we wszystkich wersjach! Każda wersja ma załączony ten sam plik pdf! Podczas

składania zamówienia upewnij się, którą wersję zamawiasz! – <http://sklep.avt.pl>

W przypadku braku dostępności na stronie sklepu osoby zainteresowane zakupem płytek drukowanych (PCB) prosimy o kontakt Via e-mail: [kity@avt.pl](mailto:kity@avt.pl)

# Kamera z ruchomym ramieniem

Moduł ESP32-CAM, czyli mikrokontroler ESP32 zintegrowany z miniaturową kamerką, cieszy się od lat zasłużoną popularnością. Za umiarkowaną cenę użytkownik dostaje urządzenie pozwalające uzyskać obraz wideo o przyzwoitych parametrach. W dodatku oprogramowanie modułu ma otwarty kod źródłowy, co pozwala dostosowywać je do swoich potrzeb i pomysłów. Zapewne dlatego wciąż powstaje wiele projektów z tym modułem. Inspiracją dla zaprezentowanego projektu była potrzeba rozruszania ESP32-CAM – co miało poszerzyć jego możliwości np. w opiece nad dziećmi czy chorymi itp.



Kamera montowana w gnieździe ZIF modułu ma niewymienny obiektyw o kącie widzenia około 25°. W niektórych serwisach internetowych oferowane są kamerki z obiektywami o szerszym kącie, który powoduje zniekształcenie obrazu, czyli tzw. efekt rybiego oka. W tym projekcie dzięki umieszczeniu modułu na ruchomym ramieniu uzyskuje się pole widzenia o zakresie 180° zarówno w płaszczyźnie horyzontalnej, jak

i wertykalnej i to dla kamerki ze standardowym obiektywem. Wygląd gotowej konstrukcji pokazuje zdjęcie tytułowe.

## Konstrukcja zdalnie sterowanego mechanizmu

W skład projektu wchodzi: ramię przegubowe z miniaturowymi serwomechanizmami, 2 płytki drukowane i moduł ESP32-CAM, który po zaprogramowaniu pracuje jako serwer sieciowy dodatkowo sterujący ruchami ramienia. Mniejsza z płytek drukowanych służy do zamocowania modułu na końcu ramienia oraz do doprowadzenia zasilania

i sygnałów do ESP32-CAM. Większa płytka pełni funkcję stabilnej podstawy dla ramienia. Znajdują się na niej gniazda do przyłączenia kabli serwomechanizmów, przewodów sygnałowych z modułu oraz elementy obwodu zasilania.

## Budowa i działanie

Schemat ideowy układu zamontowanego na obydwu płytkach pokazany został na **rysunku 1** i **rysunku 2**. Jak widać, nie jest on bardzo skomplikowany. Na schemacie z rysunku 1 pokazano połączenia elementów zamontowanych na małej płytce.

**WYKAZ ELEMENTÓW**, które możesz zamówić w sklepie AVT na stronie [sklep.avt.pl](http://sklep.avt.pl) lub bezpośrednio (ul. Leszczyńska 11, 03-197 Warszawa, tel. 48222578451, e-mail: [handlowy@avt.pl](mailto:handlowy@avt.pl)):

**Rezystory:** (SMD0805)

- R1: 1 kΩ
- R2: 30 kΩ
- R3: 50 kΩ
- R1: 2,2 kΩ

**Kondensatory:**

- C1: 10 µF SMD0805
- C2: 100 nF SMD0805
- C1: 100 nF SMD0805

- C2: 10 µF SMD0805
- C3, C4: 100 µF/25 V SMD

**Pozostałe:**

- J1, J2: DG308-2,54/3 złącze śrubowe do druku
- 2 × listwa gniazd goldpin 1×8
- K1: P-5 przekaźnik miniaturowy przewlekany
- J2: USB B-Mini gniazdo USB
- J5, J6: DG308-2,54/3 złącze śrubowe do druku
- J1: DG127-5.0/2 złącze śrubowe do druku

- J3, J3: styki kątowe goldpin 1×3 uchwyt przegubowy do kamery FPV
- 2 × serwomechanizm SG90

**Półprzewodniki:**

- D1: SMBJ5.0A dioda zabezpieczająca szybka
- M1: ESP32-CAM
- D1: 1N4007 SMD
- Q1: BC847 SMD

Do osadzenia modułu ESP32-CAM służy gniazdo M1 składające się z 2 listw gniazd goldpin o rastrze 2,54 mm. Gniazdo J1 dostarcza do modułu zasilania VCC o napięciu +5 V. Na kolejnych stykach gniazda J2 wyprowadzono:

- sygnał załączania przekaźnika doprowadzającego zasilanie do serw – J2-1,
- sygnały PWM sterujące serwomechanizmami ramienia w płaszczyźnie:
  - wertykalnej – J2-2,
  - horyzontalnej – J2-3.

Oporniki R2, R3 tworzą dzielnik obniżający wartość napięcia VCC do poziomu umożliwiającego jego podanie na wejście przetwornika ADC IO2. Szybka dioda D1 dodatkowo zabezpiecza wejście przed przepięciami. Pomiary napięcia VCC będą wykorzystywane w kolejnych wersjach oprogramowania.

Na schemacie z rysunku 2 pokazane zostały połączenia pomiędzy elementami zamontowanymi na dużej płytce. Napięcie zasilania +5 V z zewnętrznego zasilacza podawane jest na gniazdo J2 typu USB. Do gniazda podłączona jest równolegle kostka zaciskowa J1. Zasilanie można podłączyć albo do gniazda, albo do kostki.

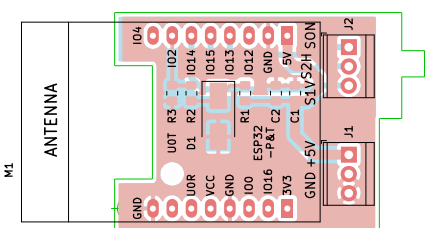
**Uwaga! Obwody zasilania nie są zabezpieczone przed omyłkową zmianą polaryzacji. Nie należy podłączać zasilania jednocześnie do gniazda i do kostki.**

Gniazdo J6 służy do podania zasilania na małą płytkę. Z kolei z małej płytki do gniazda J5 podawane są sygnały PWM sterujące serwami. Z tego samego gniazda, ze złącza J5-1 pobierany jest sygnał załączania przekaźnika K1. Przewody serwomechanizmów należy podłączyć do styków złączy J3 i J4.

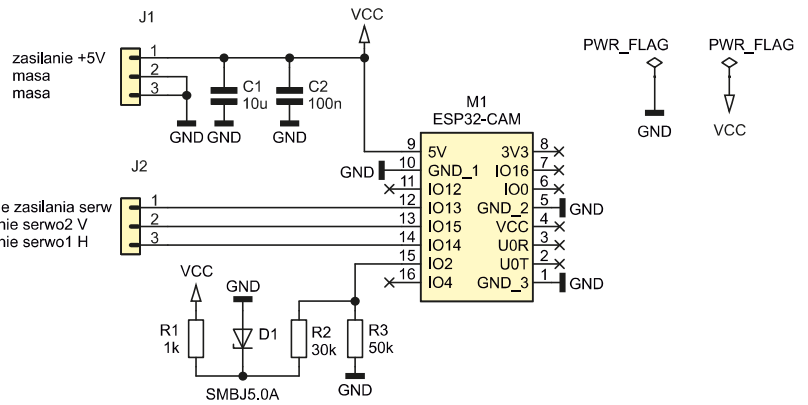
### Montaż i uruchomienie

Montaż można rozpocząć od złożenia ruchomego ramienia poruszanego miniaturowymi serwomechanizmami typu SG90 lub MG90. Ułatwieniem może być dokumentacja producenta ramienia [1].

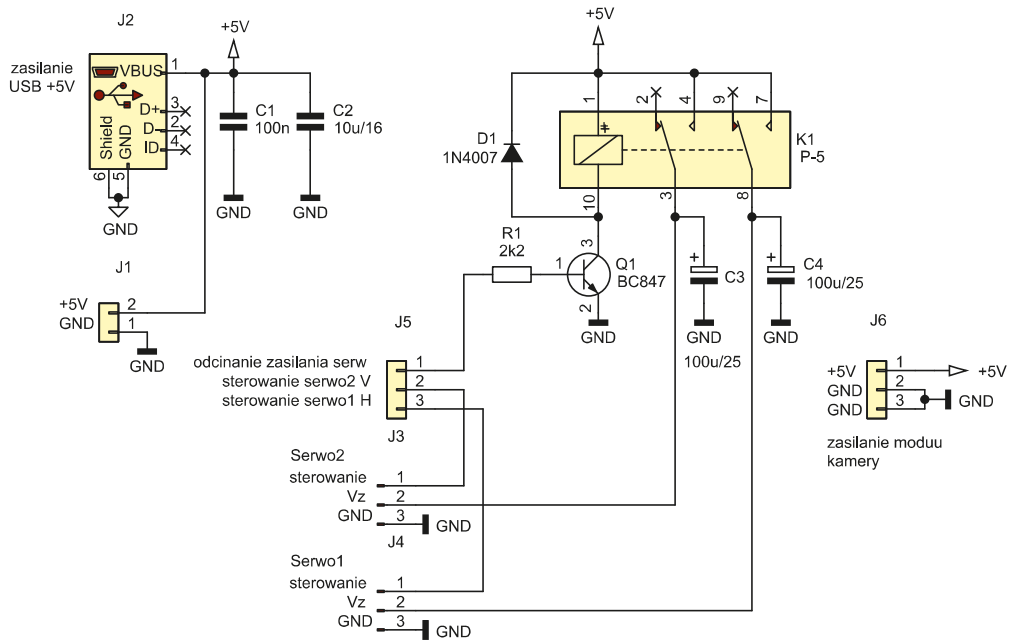
Schematy płytek PCB zostały pokazane na **rysunku 3** i na **rysunku 4**. Końcówki elementów przewlekanych przeznaczonych



Rysunek 3. Schemat płytki PCB dla modułu ESP32



Rysunek 1. Schemat układu z modułem ESP32 (mała płytka)



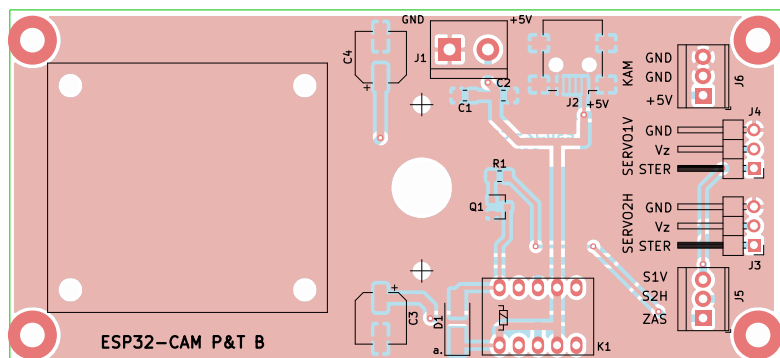
Rysunek 2. Schemat części bazowej (duża płytka)

do montażu na małej płytce należy maksymalnie skrócić i obciąć tak, żeby nie wystawały ponad pola lutownicze. Ułatwia to mocowanie płytki drukowanej, którą przykręca się jednym wkrętem do uchwyty kamery na końcu ramienia. Otwór pod wkręt należy wywiercić w plastikowej płytce uchwyty kamery. Gniazda J1, J2 trzeba wlotować otworami dla kabli na zewnątrz w kierunku brzegu płytki.

Z kolei na dużej płytce drukowanej śrubowe gniazda zaciskowe J5, J6 należy wlotować otworami na kabli do wewnątrz płytki, co ułatwi poprowadzenie przewodów (**fotografia 1**).



Fotografia 1. Mocowanie przewodów do dużej płytki



Rysunek 4. Schemat płytki PCB bazowej

To samo dotyczy złączy J3, J4 do podłączenia kabli serwo mechanizmów. Następnym krokiem jest przykręcenie podstawy ramienia do dużej płytki drukowanej. Posłużą do tego śrubki dołączone do zestawu montażowego ramienia o długości 6 mm i średnicy 2 mm.

Należy poprowadzić przewody łączące styki gniazda J1 małej płytki ze stykami gniazda J6 dużej płytki oraz styki gniazda J2 ze stykami J5. Połączone powinny być ze sobą styki gniazd o tych samych numerach. Do połączeń należy użyć cienkich kabli, które nie będą stawiać dużych oporów w czasie ruchów ramienia. W modelu użyto fragmentów szarego wielożyłowego kabla taśmy komputerowej. Każdy z odcinków powinien mieć długość 15...20 cm. Należy usunąć kilkumilimetrowe końcówki izolacji na krańcach przewodów, a druty żyły zagiąć do tyłu. Następnie zacisnąć żyłę razem z izolacją w otworach kostek gniazd śrubowych. Dzięki temu zapewniony zostanie kontakt elektryczny, a przewód nie będzie się urywał podczas ruchu ramienia.

Przewody można pozostawić swobodnie albo przed zamocowaniem w gniazdach dużej płytki zwinąć w pętle i unieruchomić poprzeczką przykręconą dwoma wkrętami (po obu stronach dużego otworu) do większej płytki. W modelu do mocowania pętli przewodów zastosowano niewykorzystaną belkę orczyka, skróconą i z nawierconymi otworami dla wkrętów mocujących.

Na koniec w rogach większej płytki można przylepić gumowe nóżki, które zapewnią lepszą stabilność całej konstrukcji.

## Oprogramowanie

Sterowanie modulem kamery odbywa się za pośrednictwem przeglądarki internetowej i stron generowanych przez sam moduł. Oprogramowanie modułu powstało jako rozszerzenie szkicu dla Arduino [2], który z kolei bazuje na przykładowym opracowaniu producenta procesora ESP32, firmy Espressif [3]. W porównaniu do pierwowzoru wprowadzono kilka zmian: zrezygnowano z funkcji rozpoznawania twarzy, dodano możliwość zdalnej aktualizacji nowych wersji oprogramowania drogą radiową (OTA), wprowadzono możliwość przywracania ustawień kamery, uporządkowano i uproszczono wygląd wyświetlanych w przeglądarce stron generowanych przez moduł ESP32-CAM. No i oczywiście dodano możliwość sterowania serwami użytymi do poruszania ramieniem.

Można wybierać pomiędzy maksymalnie uproszczonym widokiem strony sterującej (rysunek 5), rozwinięciem z dostępem zarówno do suwaków, jak i przycisków precyzyjnego ustawiania położenia ramienia z kamerą (rysunek 6) albo uzyskać dostęp do wielu nastaw kamery (rysunek 7). Przełączanie pomiędzy różnymi widokami strony z elementami sterującymi uzyskuje się po naciśnięciu elementu oznaczonego jako *Menu*.

Klawisz *Get Still* powoduje kolejne wyświetlanie kilku ostatnich stop klatek. Klawisz *Start/Stop Stream* wznawia lub wstrzymuje transmisję strumieniową. Natomiast naciśnięcie przycisku *Full* spowoduje rozwinięcie ekranu z nastawami parametrów pracy kamery. Na samym dole listy nastaw dodano napis *Servo* będący linkiem do podstrony

z ustawieniami związanymi z serwo mechanizmami. W wersji obecnej można tam zaprogramować pozycję ramienia, którą przyjmie po resecie oraz ograniczenia jego ruchów w pionie i poziomie.

Dodatkową opcją jest możliwość sterowania kamerą i ramieniem za pomocą rozkazów wpisywanych bezpośrednio w pasku przeglądarki i przesyłanych w trybie *GET*. Pozwala to tworzyć skrypty i zautomatyzować proces sterowania. Listę rozkazów można znaleźć tutaj [4].

Pliki źródłowe oprogramowania modułu dla środowiska Arduino można pobrać tu [5].

Przed zaprogramowaniem modułu zaleca się przeprowadzić kompilację z własnymi ustawieniami predefiniowanymi dla wybranego typu modułu. Dalszy opis dotyczyć będzie ESP32-CAM AI-THINKER z kontrolerem w wersji ESP32-S z dostępem do dodatkowej zamontowanej pamięci PSRAM.

Żeby móc przeprowadzić kompilację, należy w środowisku Arduino mieć zainstalowaną obsługę modułów z procesorem ESP32. Jeżeli jeszcze tego nie zrobiłeś, dodaj dodatkowy wpis w polu

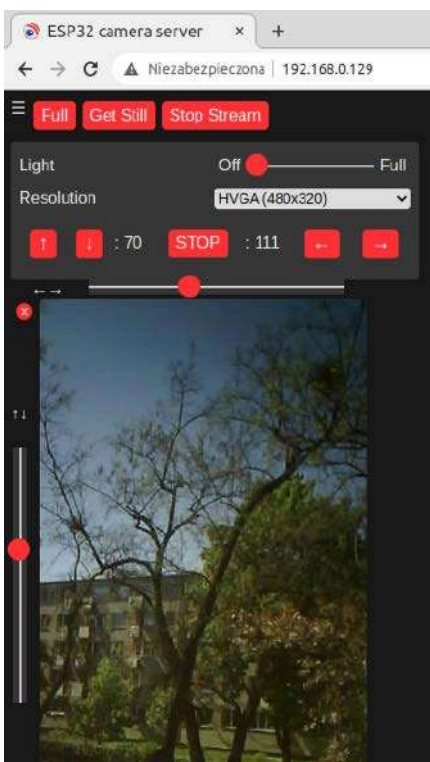
File → Preferences → Additional Board Manager URLs:

[https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package\\_esp32\\_index.json](https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json)

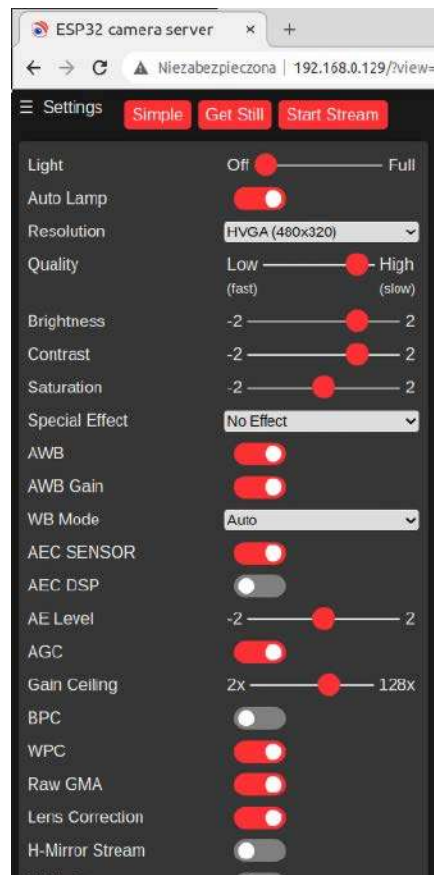
Następnie przejdź do Tools → Boards → ESP32 Arduino i wybierz ESP32 Dev Module. Dla płytki ustaw parametry:



Rysunek 5. Uproszczony wygląd pulpitu sterującego



Rysunek 6. Pulpit z elementami sterującymi ruchem ramienia



Rysunek 7. Ustawianie parametrów kamery



- CPU Frequency: 240 MHz
- Flash Mode: QIO
- Flash Size: 4 MB
- Partition Scheme: Minimal SPIFFS (1.9 MB APP with OTA/190 KB SPIFFS)
- PSRAM: Enabled

Wybierz w ustawieniach port USB, do którego podłączony jest interfejs USB-Rx/Tx za pomocą którego zaprogramujesz moduł ESP32-CAM. Pamiętaj, żeby podłączyć:

- wyjście interfejsu Tx z UOR (GPIO3) modułu,
- wejście interfejsu Rx z UOT (GPIO1) modułu,
- zwrzyj wyprowadzenie GPIO0 modułu z GND,
- doprowadź do wejścia 5 V napięcie zasilania +5 V o wydajności prądowej zasilacza co najmniej 500 mA.

Pamiętaj, żeby wyprowadzenia Tx i Rx interfejsu pracowały z sygnałami o maksymalnych poziomach 3,3 V. To ważne gdyż praca z wyższymi poziomami może uszkodzić moduł.

Zanim przejdziesz do kompilacji, edytuj plik *myconfig.h* lub utwórz nowy o takiej nazwie, korzystając z szablonu *myconfig.sample.h*. Jest to plik z predefiniowanymi ustawieniami użytkownika. Uzupełnij strukturę `struct station stationList[]`, wpisując nazwę sieci (ssid) i hasło dostępu (password). Można podać 2 pary parametrów dostępu do jednej sieci lub więcej.

Po włączeniu zasilania moduł zacznie szukać sieci o nazwie z pozycji 2 lub kolejnych. Jeżeli żadnej nie znajdzie i nie będzie się mógł zalogować, automatycznie przejdzie w tryb

pracy punktu dostępowego i utworzy własną sieć o nazwie i hasle z pierwszego miejsca listy. Żeby tak się stało, należy odkomentować pozycję

```
#define WIFI_AP_ENABLE
```

Możesz także zmienić domyślny adres IP punktu dostępowego, edytując pozycję

```
#define AP_ADDRESS 192,168,4,1
```

(zwróć uwagę, że kolejne pozycje adresu IP w tej definicji są oddzielone znakiem przecinka).

Używając modułu AI-THINKER, pamiętaj, aby linia

```
#define CAMERA_MODEL_AI_THINKER
```

została odkomentowana.

Prawdopodobnie przed kompilacją będziesz musiał w zwykły dla Arduino sposób doinstalować niektóre z wymienionych poniżej bibliotek:

```
AsyncTCP
ESPAsyncWebServer
AsyncElegantOTA
uTimerLib
```

W przypadku gdy moduł ESP32-CAM zalogował się do sieci, o jego numer IP w jej zasięgu należy odpytywać router zarządzający siecią. Po pierwszym zaprogramowaniu modułu kolejne aktualizacje oprogramowania można już przeprowadzać zdalnie, drogą radiową korzystając z trybu OTA. Najpierw należy odpowiednio skompilować źródła oprogramowania, korzystając z opcji Sketch → Export compile binary. Po zakończeniu kompilacji w pasku przeglądarki wpisujemy adres IP modułu uzupełniony o numer portu 83 i rozkaz update np. tak:

```
http://192.168.0.129:83/update
```

Gdy zgłosi się strona Elegant OTA, należy wybrać opcję Firmware i wskazać lokalizację wygenerowanego podczas kompilacji pliku binarnego (jeżeli korzystamy ze szkicu Arduino o nazwie *esp32-cam-pan-tilt.ino*, plik będzie się nazywał *esp32-cam-pan-tilt.ino.esp32.bin*). Gdy wyświetli się informacja o sukcesie zdalnego zapisu, należy nacisnąć przycisk *Back* i w pasku przeglądarki wpisać IP modułu bez numeru portu.

## Bilans prądów

Moduł wraz z ramieniem zależnie od trybu pracy mogą pobierać różny, czasem znacząco wyższy prąd. Poniżej znajdują się wartości pomiarów dla różnych sytuacji przy zasilaniu napięciem 5 V:

- serwa w trybie oczekiwania: 6 mA,
- 1 serwo stabilne 2 pracuje: 12 mA,
- moduł transmituje strumień obrazów: ok. 180 mA,
- wstrzymana transmisja obrazu: 170 mA,
- transmisja obrazu, lampa LED włączona na max.: 330 mA.

Z pomiarów wynika, że zewnętrzny zasilacz powinien dostarczać stabilnego napięcia zasilania +5 V przy wydajności co najmniej 500 mA.

**Ryszard Szymaniak**  
biuro@ars.info.pl

[1] <https://bit.ly/3HIN7oC>

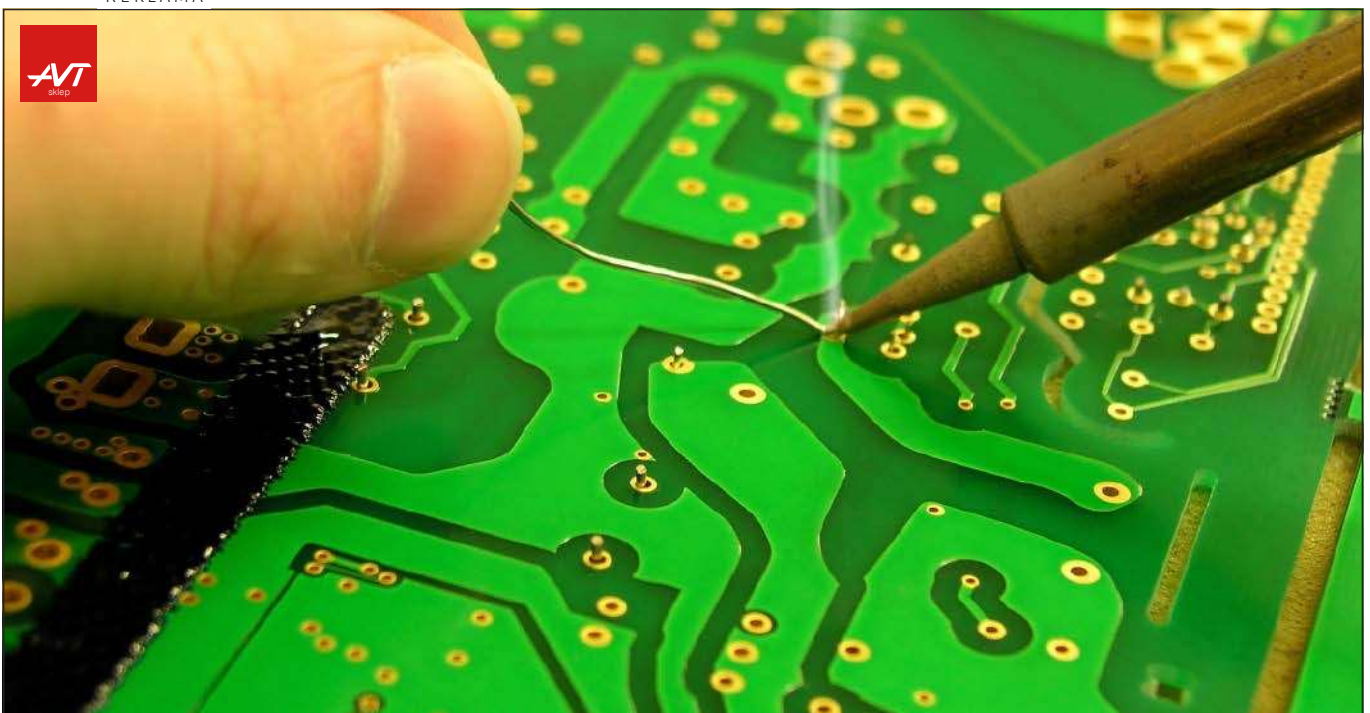
[2] <https://bit.ly/3yclMrv>

[3] <https://bit.ly/3xR4Bux>

[4] <https://bit.ly/3xNOFZR>

[5] <https://bit.ly/3ybyy9M>

REKLAMA



**KITY AVT**

@KITYAVT • Elektronika • <http://bit.ly/2BjVMN7>