

# Asystent cyfrowy ZW-1

*Prezentowane urządzenie łączy moc obliczeniową komputerka Raspberry Pi Zero W z kilkoma komponentami umożliwiającymi interakcje z otoczeniem. Całość programowana jest za pomocą języka Python, dzięki czemu ma naprawdę niebagatelne możliwości.*

Urządzenie przypomina nietypową płytkę ewaluacyjną. Oferuje zasoby sprzętowe ukierunkowane na realizację interfejsu użytkownika i monitorowanie otoczenia oraz program inicjujący te zasoby i pokazujący, jak z nich korzystać we własnych programach. Natomiast docelowa funkcja urządzenia zależy od kreatywności konstruktora/programisty. Najważniejsze funkcje i możliwości asystenta cyfrowego to:

- odtwarzanie komunikatów głosowych,
- sygnalizacja dźwiękowa za pomocą buzzera,
- wyświetlanie krótkich treści na wyświetlaczu LCD 2×16,
- oświetlanie otoczenia za pomocą diod LED,
- pomiar temperatury otoczenia,
- odczytywanie stanu 3 przycisków,
- wykrywanie ruchu za pomocą czujnika PIR,
- możliwość połączenia z siecią Wi-Fi,

- pozwala obserwować otoczenie za pomocą kamery.

## Budowa urządzenia

Budowę Asystenta cyfrowego ZW-1 można podzielić na dwie części. Pierwsza część to komputer Raspberry Pi Zero W oraz komponenty niezbędne do jego działania i rozszerzające jego funkcje multimedialne:

- kamera dla Raspberry Pi 8 Mpx + taśma połączeniowa do kamery o długości 150 mm,
- karta dźwiękowa USB + przewód OTG microUSB ↔ USB o długości 120 mm,
- karta pamięci microSD o pojemności 16 GB,
- zasilacz mikroUSB 5 V o wydajności prądowej 2 A.

Połączone komponenty pierwszej części urządzenia pokazuje **fotografia 1**.

Druga część to płytka drukowana, na której znajdują się elementy elektroniczne, dzięki którym asystent cyfrowy ZW-1 komunikuje się z użytkownikiem oraz pobiera dane z otoczenia. Schemat ideowy części elektronicznej został pokazany na **rysunku 1**. Złącze J1 służy do podłączenia komputera Raspberry Pi Zero W. Układ scalony oznaczony U1 oraz współpracujące z nim elementy dyskretne C1, C2, C3 oraz R1 mają za zadanie wzmacniać sygnał audio pochodzący z karty dźwiękowej. Wzmocniony sygnał trafia na złącze J2

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

**W ofercie AVT\* AVT-----**

### Podstawowe parametry:

- bazuje na komputerku Raspberry Pi Zero W,
- zawiera komponenty niezbędne do monitorowania otoczenia i realizacji interfejsu użytkownika,
- umożliwia łączność za pomocą Wi-Fi,
- programowany za pomocą języka Python.

### Projekty pokrewne na [www.media.avt.pl](http://www.media.avt.pl):

----	Kieszonkowy Linux (EP 9/2020)
AVT-5776	Miniaturowa czujka ruchu dla Raspberry Pi, Arduino i nie tylko (EP 6/2020)
AVT-5770	Arduino i nie tylko (EP 5/2020)
AVT-5761	Czterokanałowy moduł przekaźnikowy sterowany I <sup>2</sup> C (EP 4/2020)
Projekt 249	Karta muzyczna dla Raspberry Pi (EP 3/2020)
AVT-5739	Zasilacz buforowy z superkondensatorami dla Raspberry Pi (EP 1/2020)
----	Instalacja zegara czasu rzeczywistego w Raspberry Pi (EP 9/2019)
----	Domowy serwer Network Attached Storage na Raspberry Pi (EP 9/2019)
AVT-5689	Budżetowy interfejs USB-I <sup>2</sup> S (EP 7/2019)
AVT-5680	Sensor smogu z technologią Bluetooth Low Energy (EP 4/2019)

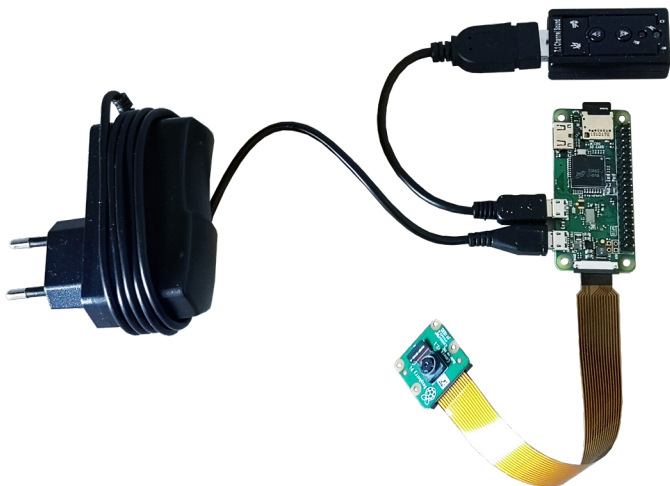
### Uwaga! Elektroniczne zestawy do samodzielnego montażu.

#### Wymagana umiejętność lutowni!

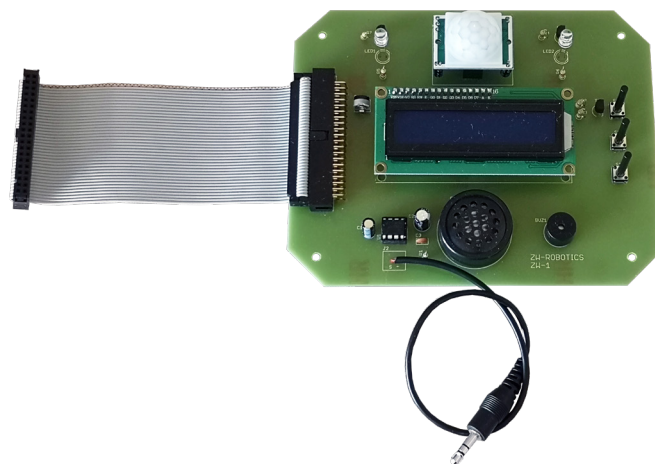
Podstawową wersją zestawu jest wersja [B] nazywana potocznie KIT-em (z ang. zestaw). Zestaw w wersji [B] zawiera elementy elektroniczne (w tym [UK] – jeśli występuje w projekcie), które należy samodzielnie wzlutować w dołączoną płytkę drukowaną (PCB). Wykaz elementów znajduje się w dokumentacji, która jest podlinkowana w opisie kitu.

- Mając na uwadze różne potrzeby naszych klientów, oferujemy dodatkowe wersje:
- wersja [C] – zmontowany, uruchomiony i przetestowany zestaw [B] (elementy wzlutowane w płytkę PCB)
  - wersja [A] – płytka drukowana bez elementów i dokumentacji 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 <http://sklep.avt.pl>, osoby zainteresowane zakupem płytek drukowanych (PCB) prosimy o kontakt via e-mail: [kity@avt.pl](mailto:kity@avt.pl).



Fotografia 1. Komputer Raspberry Pi Zero W oraz komponenty niezbędne do jego działania i rozszerzające jego funkcje multimedialne



Fotografia 2. Wygląd zmontowanej płytki drukowanej z wyprowadzonymi przewodami do podłączenia komputera Raspberry Pi Zero W oraz karty dźwiękowej

i jest następnie odtwarzany w głośniku G1. Za pomocą potencjometru montażowego P1 jest ustawiany kontrast wyświetlacza LCD1, a elementy elektroniczne T1, R2 oraz R3 są odpowiedzialne za sterowanie jego podświetlaniem. Przyciski S1, S2 i S3 są dołączone do wyprowadzeń GPIO komputera, ich funkcję można dowolnie zaprogramować. Czujnik U2 typu DS18B20 umożliwia pomiar temperatury otoczenia. Rezystor R4 dołączony do wyjścia czujnika, magistrali 1-Wire, realizuje podciąganie do potencjału dodatniego. Złącze oznaczone MOD2 służy dołączeniu gotowego modułu czujnika ruchu. Moduł ten ma dwa potencjometry, dzięki którym można ustawić czas trwania stanu wysokiego na jego wyjściu po wykryciu ruchu, oraz zasięg jego działania.

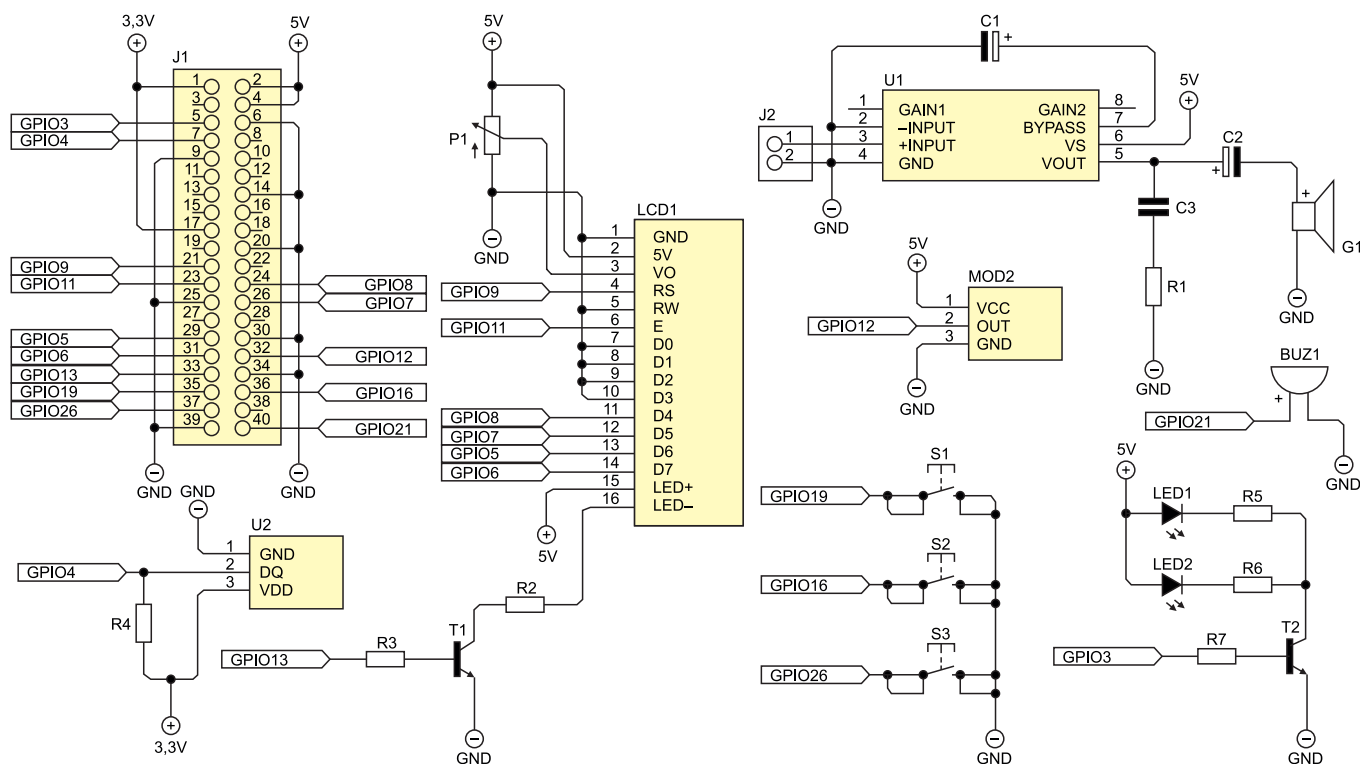
Elementy R7 i T2 sterują diodami LED1 i LED2. Prąd płynący przez diody ograniczają rezystory R5 i R6. Element BUZ1 to sygnalizator dźwiękowy z generatorem, sterowany jest bezpośrednio z wyprowadzenia GPIO21 komputerka.

### Montaż i uruchomienie

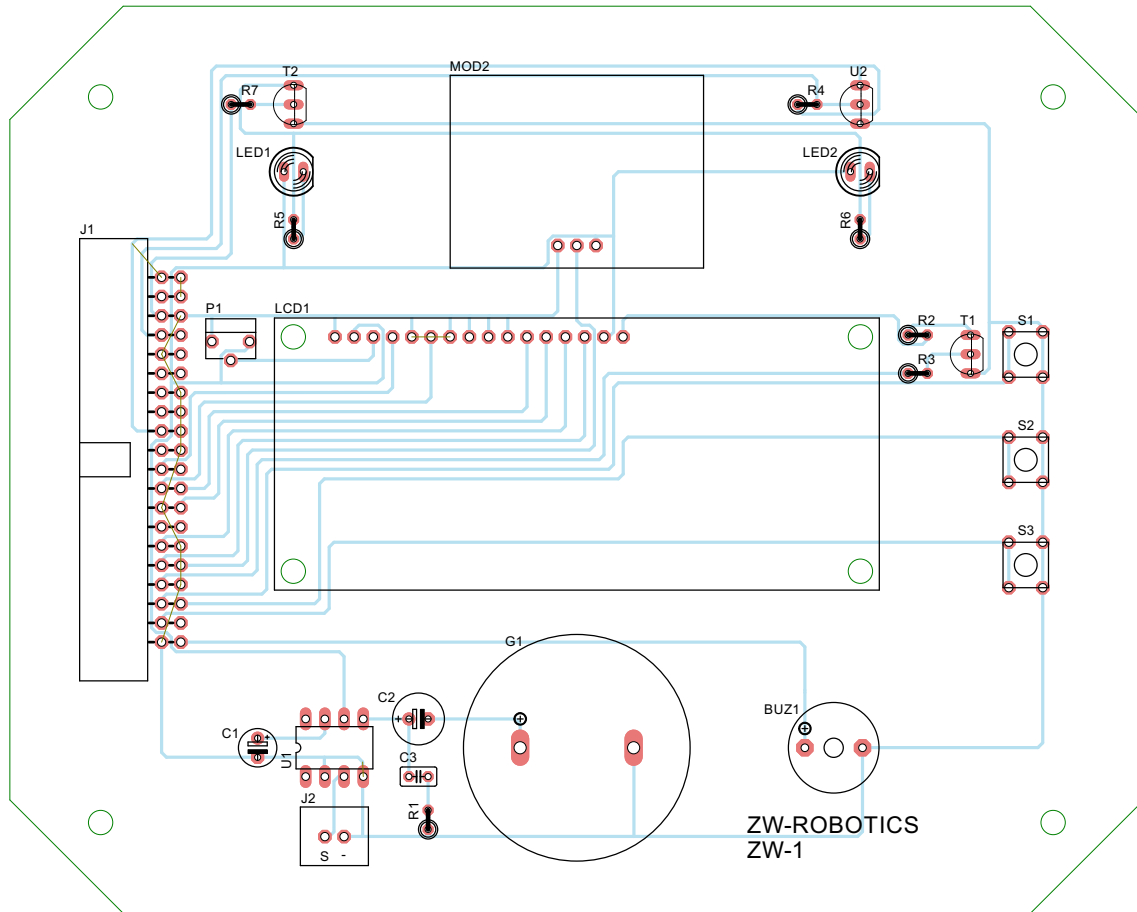
Na **rysunku 2** został pokazany schemat płytki drukowanej wraz z rozmieszczeniem elementów. Zmontowana płytka z wyprowadzonymi przewodami do podłączenia komputerka Raspberry Pi Zero W oraz karty dźwiękowej została pokazana na **fotografii 2**.

Konstrukcja mechaniczna asystenta cyfrowego ZW-1 jest wykonana z czarnego szkła akrylowego oraz niewielkiej liczby elementów montażowych. Do tylnej ścianki

obudowy przymocowano komputerkę Raspberry Pi Zero W za pomocą tulei dystansowych M3×6 mm o gwincie obustronnie wewnętrznym. Do zewnętrznych otworów przykręcono dłuższe tuleje dystansowe M3×30 mm o gwincie obustronnie wewnętrznym, będą one potrzebne do montażu płytki drukowanej. Kartę dźwiękową przyklejono z wykorzystaniem rzepa samoprzylepnego o wymiarach ok. 25×25 mm (**fotografia 3**). Za pomocą tulei dystansowych M3×15 mm o gwincie obustronnie wewnętrznym połączono wyświetlacz LCD z płytką, a następnie płytkę przykręcono za pomocą tulei dystansowych M3×20 mm o gwincie wewnętrznym oraz zewnętrznym do przygotowanej wcześniej ścianki tylnej obudowy. Do górnej części obudowy przykręcono czujnik ruchu



Rysunek 1. Schemat ideowy układu elektronicznego



Rysunek 2. Schemat płytki drukowanej z rozmieszczeniem elementów

i kamerę za pomocą śrub oraz nakrętek M2 (fotografia 4). Na zakończenie montażu konstrukcji mechanicznej asystenta cyfrowego ZW-1, uchwytami LED RTC51 stabilnie przymocowano diody LED oraz przykręcono górną część obudowy.

- Wykaz elementów:**
- Rezystory:**  
 R1: 10 Ω  
 R2: 33 Ω  
 R3, R4, R7: 4,7 kΩ  
 R5, R6: 220 Ω  
 P1: potencjometr montażowy 10 kΩ
- Kondensatory:**  
 C1: 10 μF/25 V  
 C2: 220 μF/25 V  
 C3: 47 nF ceramiczny
- Półprzewodniki:**  
 LED1, LED2: dioda LED 5 mm biała  
 T1, T2: BC337  
 U1: LM386L DIP8  
 U2: DS18B20
- Pozostałe:**  
 S1..S3: przycisk 6x6 mm o długości 22,5 mm  
 G1: głośnik 8 Ω, 0,5 W, o średnicy 30 mm  
 BUZ1: buzzer z generatorem 3 V  
 J1: gniazdo IDC 40 pin + wtyk IDC kątowy 40 pin + przewód taśmowy o długości 140 mm  
 LCD1: wyświetlacz LCD 2x16 HD44780  
 MOD2: czujnik ruchu PIR HC-SR501  
 wtyk jack 3,5 z przewodem o długości 240 mm  
 Raspberry Pi Zero W + osprzęt (wykaz w tekście)  
 elementy mocujące - tuleje, śruby, nakrętki, itd.

**Listing 1. Kod programu prezentującego możliwości urządzenia**

```
import os
from time import sleep
import Adafruit_CharLCD as LCD
import w1thermsensor
from picamera import PiCamera
import RPi.GPIO as GPIO

GPIO.setmode(GPIO.BCM)

GPIO.setup(3, GPIO.OUT)
GPIO.setup(13, GPIO.OUT)
GPIO.setup(21, GPIO.OUT)
GPIO.setup(12, GPIO.IN)
GPIO.setup(16, GPIO.IN, pull_up_down=GPIO.PUD_UP)
GPIO.setup(19, GPIO.IN, pull_up_down=GPIO.PUD_UP)
GPIO.setup(26, GPIO.IN, pull_up_down=GPIO.PUD_UP)

lcd = LCD.Adafruit_CharLCD(9, 11, 8, 7, 5, 6, 16, 2)
lcd.create_char(1,[7, 5, 7, 32, 32, 32, 32, 32])
lcd.clear()

def komunikat_glosowy(tekst):
    os.system("espeak "+tekst+" --stdout -a 200 -s 180 -p 40 | aplay 2>/dev/null' )

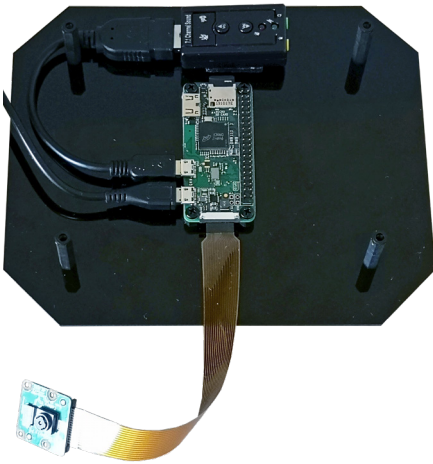
Zmienna_pomocnicza1 = True
Zmienna_pomocnicza2 = True

Komunikat_glosowy("Witaj, Nazywam się ZW-1,
jestem asystentem cyfrowym,
stworzył mnie Zygmunt Wypich.
Posiadam wiele umiejętności.")

sleep(2)
Komunikat_glosowy("Informuję o zdarzeniach za pomocą buzzera.")
sleep(2)
GPIO.output(21, GPIO.HIGH)
sleep(2)
GPIO.output(21, GPIO.LOW)

sleep(2)
Komunikat_glosowy("Wyświetlam dane na wyświetlaczu LCD.")
sleep(2)
GPIO.output(13, GPIO.HIGH)
lcd.message('Witaj!')
sleep(7)
lcd.clear()
GPIO.output(13, GPIO.LOW)

sleep(2)
Komunikat_glosowy("Oświetlam otoczenie za pomocą diod LED.")
sleep(2)
GPIO.output(3, GPIO.HIGH)
sleep(7)
```



Fotografia 3. Przygotowanie tylnej ścianki obudowy

## Oprogramowanie

Przed rozpoczęciem programowania asystenta cyfrowego ZW-1 konieczne jest zainstalowanie bibliotek języka Python do obsługi wyświetlacza LCD, czujnika temperatury oraz syntezy mowy. W tym celu łączymy się z komputerkiem np. za pomocą PuTTY i wpisujemy następujące polecenia:

```
pi@raspberrypi:~ $ git
clone https://github.com/adafruit/Adafruit_Python_CharLCD.git
pi@raspberrypi:~ $ cd
Adafruit_Python_CharLCD
pi@raspberrypi:~ / Adafruit_Python_CharLCD $ sudo python3
setup.py install
pi@raspberrypi:~ $ pip3 install
w1thermsensor
```

### Listing 1. cd.

```
GPIO.output(3, GPIO.LOW)

sleep(2)
Komunikat_glosowy("Mierzę aktualną temperaturę otoczenia.")
sleep(2)
GPIO.output(13, GPIO.HIGH)
lcd.clear()
sensor = w1thermsensor.W1ThermSensor()
temp = sensor.get_temperature()
temp = round(temp, 1)
temperatura = str(temp)
lcd.message('Temperatura:')
lcd.set_cursor(0,1)
lcd.message(temperatura + '\x01' + 'C')
sleep(7)
lcd.clear()
GPIO.output(13, GPIO.LOW)

sleep(2)
Komunikat_glosowy("Odczytuję stan trzech przycisków ogólnego przeznaczenia.")
sleep(1)
while (Zmienna_pomocnicza1 == True):
    if GPIO.input(19) == False and Zmienna_pomocnicza2 == True:
        sleep(0.2)
        Komunikat_glosowy("Został naciśnięty górny przycisk.")
        Zmienna_pomocnicza1 = False
        Zmienna_pomocnicza2 = GPIO.input(19)

sleep(1)
Zmienna_pomocnicza1 = True
Zmienna_pomocnicza2 = True
while (Zmienna_pomocnicza1 == True):
    if GPIO.input(16) == False and Zmienna_pomocnicza2 == True:
        sleep(0.2)
        Komunikat_glosowy("Został naciśnięty środkowy przycisk.")
        Zmienna_pomocnicza1 = False
        Zmienna_pomocnicza2 = GPIO.input(16)

sleep(1)
Zmienna_pomocnicza1 = True
Zmienna_pomocnicza2 = True
while (Zmienna_pomocnicza1 == True):
    if GPIO.input(26) == False and Zmienna_pomocnicza2 == True:
        sleep(0.2)
        Komunikat_glosowy("Został naciśnięty dolny przycisk.")
        Zmienna_pomocnicza1 = False
        Zmienna_pomocnicza2 = GPIO.input(26)

sleep(2)
Zmienna_pomocnicza1 = True
Komunikat_glosowy("Wykrywam ruch w swoim otoczeniu.")
sleep(2)
Komunikat_glosowy("Proszę opuść pomieszczenie na dziesięć sekund, a następnie ponownie się w nim pojaw. Kiedy wykryje twój ruch, poinformuję cię o tym.")
sleep(10)
while (Zmienna_pomocnicza1 == True):
    if GPIO.input(12) == True:
        Komunikat_glosowy("Twój ruch został wykryty.")
        Zmienna_pomocnicza1 = False

sleep(2)
Komunikat_glosowy("Obserwuję otoczenie za pomocą kamery.")
sleep(2)
Komunikat_glosowy("Za pięć sekund zaświecę diody LED, a następnie zrobię zdjęcie, które zapiszę na pulpicie.")
sleep(5)
GPIO.output(3, GPIO.HIGH)
Kamera = PiCamera()
Kamera.start_preview()
Kamera.capture('/home/pi/Desktop/Zdjecie.jpg')
Kamera.stop_preview()
sleep(5)
GPIO.output(3, GPIO.LOW)

sleep(2)
Komunikat_glosowy("Dziękuję za uwagę.")
```

```
pi@raspberrypi:~ $ sudo apt-get
install espeak
```

Aby syntezy mowy prawidłowo wymawiał polskie litery, należy otworzyć plik, w którym zapisane są odpowiednie ustawienia, poleceniem

```
pi@raspberrypi:~ $ sudo nano /
ust/lib/arm-linux-gnueabi/hf/
espeak-data/voices/default
a następnie zmodyfikować treść otwartego pliku:
name default
language pl
gender male
intonation 2
```

Teraz można uruchomić program prezentujący umiejętności urządzenia, którego kod został pokazany na **listingu 1**. Więcej informacji na temat asystenta cyfrowego ZW-1 oraz film prezentujący jego umiejętności znajduje się na stronie internetowej [www.zw-robotics.pl](http://www.zw-robotics.pl).

Zygmunt Wypich  
zygmunt.wypich@zw-robotics.pl

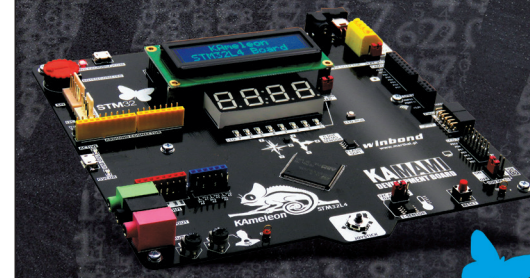


Fotografia 4. Montaż płytki drukowanej oraz przygotowanie górnej części obudowy

REKLAMA

**Kameleon**   
**STM32L4 Board**

Kompletna platforma do nauki programowania



[www.kameleonboard.org](http://www.kameleonboard.org) STM32