

Zastosowanie modułu Wi-Fi ESP-12 (3)

Bezprzewodowy czujnik stacji pogodowej

Telefon komórkowy jest jednym z tych przedmiotów, po które współcześnie – przynajmniej w naszej części świata – człowiek sięga najczęściej. A więc gdy użytkownik ma cały czas telefon komórkowy przy sobie, czemu by nie połączyć go z elementami inteligentnego domu, które mogą komunikować się w paśmie 2,4 GHz w standardzie Wi-Fi? W artykule zaprezentowano zastosowanie smartfona jako do prezentacji danych z bezprzewodowego czujnika pogody.

Czujnik mierzy trzy parametry fizyczne: temperaturę, wilgotność i ciśnienie. System, który zostanie tutaj zaprezentowany opiera się o jeden nadajnik danych oraz smartphona z aplikacją do odbioru tych danych. Nic jednak nie stoi na przeszkodzie, aby system rozbudować o kolejne czujniki i zbudować w ten sposób system informacji o parametrach atmosfery wewnątrz i na zewnątrz domu, mieszkania, lub dowolnego obiektu.

Schemat ideowy połączeń czujnika pokazano na rysunku 1. Zastosowane układy DHT11 oraz BMP180 pozwalają na odczyt podstawowych parametrów otoczenia. Są to łatwo dostępne i tanie części, przez co mają pewne ograniczenia. W wypadku DHT11 jest możliwe użycie „brata”, czyli DHT22 – układu o lepszych parametrach. Są one kompatybilne pod względem rozmieszczenia wyprowadzeń. Należy jedynie zmienić typ czujnika w kodzie programu. Podstawowe parametry sensorów wymieniono w ramce.

Moduł ESP-12 ma wyprowadzenia interfejsu szeregowego I²C. Są to piny numer 5 (SCL) i 4 (SDA). Interfejs

I²C zgodnie ze swoją zasadą działania (wyjścia typu OD lub OC, iloczyn „na drucie”) wymaga zastosowania rezystorów podciągających. Kiedy tych rezystorów zabraknie, to ze względu na obecność rezystorów podciągających w strukturze mikrokontrolera komunikacja na krótkich odcinkach przypuszczalnie będzie działać bez zarzutu, jednak ze względu na sporą stałą czasową obwodu RC utworzonego przez rezystor pull-up mikrokontrolera i pojemność przewodu połączeniowego, ani nie da się osiągnąć zbyt dużej prędkości transmisji, ani dołączyć zbyt wielu układów do magistrali. Mogą też występować błędy transmisji ze względu na zbyt łagodne zbocza sygnału SCL. Dlatego zastosowano rezystory podciągające (zasilające) 4,7 kΩ.

Czujnik DHT11 komunikuje się za pomocą jednej linii opisanej DATA. Wymaga ona zewnętrznego rezystora podciągającego o oporności 10 kΩ. W dokumentacji czujnika jest opisane dokładnie, jak przebiega odczyt wartości. Społeczność Arduino udostępnia biblioteki do obsługi obu wspomnianych sensorów, przez co ich obsługa za pomocą predefiniowanego API jest nie nastrożona trudności i bardzo łatwo jest odszukać oraz wyeliminować błędy w kodzie.

Główny moduł Wi-Fi może działać w trybie oszczędnym lub stale oczekiwać na pakiety UDP poprzez tzw.

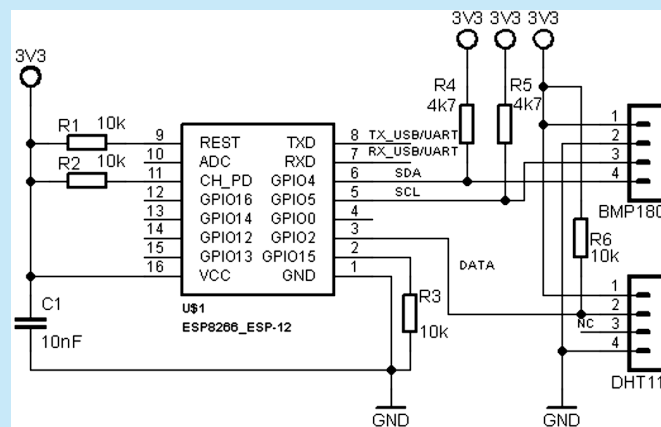
Podstawowe parametry sensorów DHT11 i BMP180

BMP180:

Napięcie zasilania: 1,8...3,6 V.
Zakres pomiarowy ciśnienia: 300...1100 hPa.
Rozdzielczość pomiarowa: 0,02 hPa.
Interfejs komunikacyjny: I²C.
Wbudowany termometr.
Układ oparty jest na technologii piezozystancyjnej.

DHT11:

Napięcie zasilania: 3...5 V.
Zakres pomiarowy temperatury: 0...50°C.
Rozdzielczość pomiaru temperatury: 8 bitów.
Dokładność pomiaru temperatury: ±1°C.
Czas odpowiedzi: 6...15 s (typowo 10 s).
Zakres pomiarowy wilgotności: 20...90%RH.
Rozdzielczość pomiarowa wilgotności: 8-bitów.
Dokładność pomiaru wilgotności: ±4 (przy 25°C).
RH – Wilgotność względna wyrażana w procentach. Jest to stosunek rzeczywistej wilgoci w powietrzu do maksymalnej jej ilości, którą może utrzymać powietrze w danej temperaturze.



Rysunek 1. Schemat ideowy połączeń czujnika

„pooling”. Jeżeli źródłem zasilania będzie bateria lub jakiś akumulator, tryb oszczędny zdecydowanie wydłuży działanie urządzenia. W opisywanym wypadku, moduł czeka na pakiety w drugi wymieniony sposób, czyli „pooling”. W efekcie, kod jest łatwiejszy do analizy oraz mniej skomplikowany. Zestaw jest stroną odpytowaną, czeka na ruch ze strony aplikacji na smartphone i dopiero wtedy podejmuje jakieś działania.

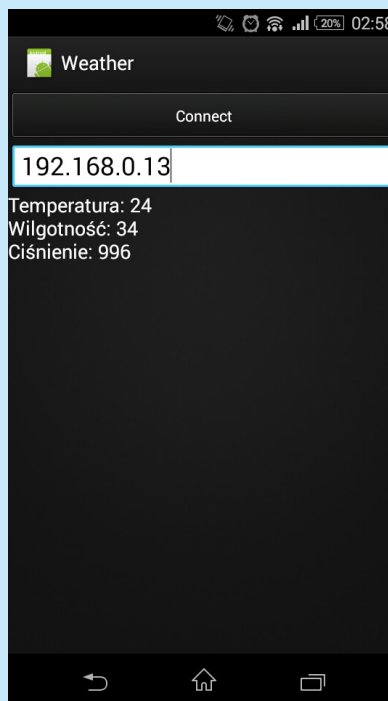
Sensory poprawnie pracują przy zasilaniu napięciem 3,3 V, więc bez problemu można je zasilic z jednego źródła wspólnie z modulem ESP-12. Energooszczędność takiego zestawu przy odpowiednim oprogramowaniu modułu umożliwia zasilanie z baterii, więc czujnik może być umieszczony w dowolnym miejscu. Jedynym warunkiem jest zasięg sieci, za której pomocą aplikacje będą się porozumiewały.

Komunikacja pomiędzy telefonem oraz zestawem odbywa się na zasadzie *Request* <-> *Response*. Odbieraniem odpowiedzi zajmuje się aplikacja stworzona na system Android w wersji 4.1 lub nowszej. Aby zainstalować aplikację należy w ustawieniach zezwolić na instalowanie aplikacji spoza sklepu Google Play. Po instalacji należy w wolne pole wpisać IP a następnie nacisnąć *connect*, po chwili wartości powinny pojawić się na środku ekranu. Adres IP nawet po wyjściu z programu jest zapisywany, więc wystarczy tylko uruchomić aplikację, aby sprawdzić obecne wartości temperatury czy ciśnienia. Działanie tej aplikacji opiera się na wysłaniu na podany adres rozkazu „meas” a następnie oczekiwaniu na odpowiedź. Odpowiedź z danymi ma formę:

Start:temperatura:wilgotność:ciśnienie:stop

Następnie, w odstępach 10-sekundowych jest powtarzany rozkaz odczytu wartości. Odebrane pomiary wyświetlają się na ekranie w postaci:

- Temperatura: 25
- Wilgotność: 34



Rysunek 2. Wygląd aplikacji dla systemu Android

- Ciśnienie: 960

Sama aplikacja jest nieskomplikowana i wykonana specjalnie na potrzeby tego artykułu (rysunek 2). Pokazuje tylko jak łatwo wykorzystać telefon jako centrum domowego systemu pomiarowego. Po lekkich modyfikacjach, aplikacja mogła by odczytywać takie wartości jak, stan czujek alarmowych, ilość domowników, parametry otoczenia, stany lamp oraz włączonych urządzeń.

Jakub Kisiel

www.microgeek.eu

REKLAMA



PRZEWODNIK PO ŚWIECIE
POKEMON GO
Z Ł A P J E W S Z Y S T K I E !

JUŻ W SPRZEDAŻY!

SZUKAJ W SIECI EMPIK I NAJLEPSZYCH SALONACH PRASOWYCH
PRZEJRZYJ I ZAMÓW NA ULUBIONYKIOSK.PL