

Modul rozpoznávání hlasu V3

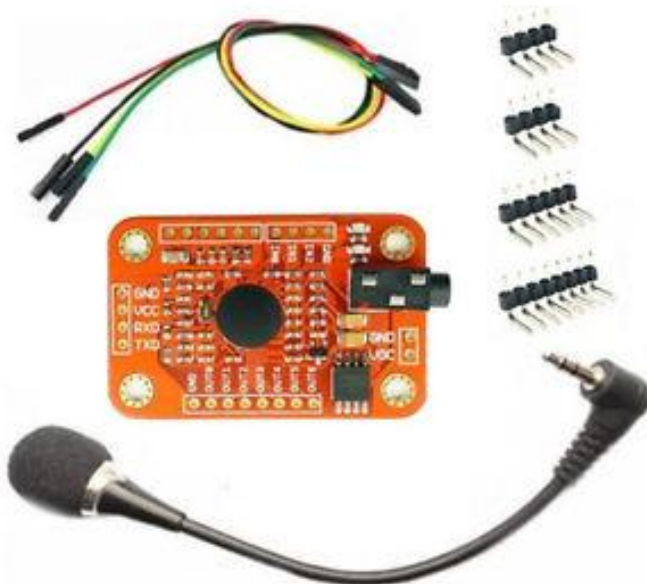


POPIS

Díky modulu lze pomocí hlasu nebo jiných zvuků ovládat různé periferie či posílat data do mikrokontroléru. Na základě těchto dat vykoná mikrokontrolér (např. Arduino) nějakou akci. Zařízení disponuje pamětí až pro 80 hlasových příkazů, přičemž každý záznam má délku 1,5 s. Hlasové příkazy jsou přidávány prostřednictvím učícího režimu.

Základní charakteristika:

- 7 výstupů
- 3 vstupy
- UART rozhraní pro komunikaci s mikrokontroléry
- mikrofon součástí balení

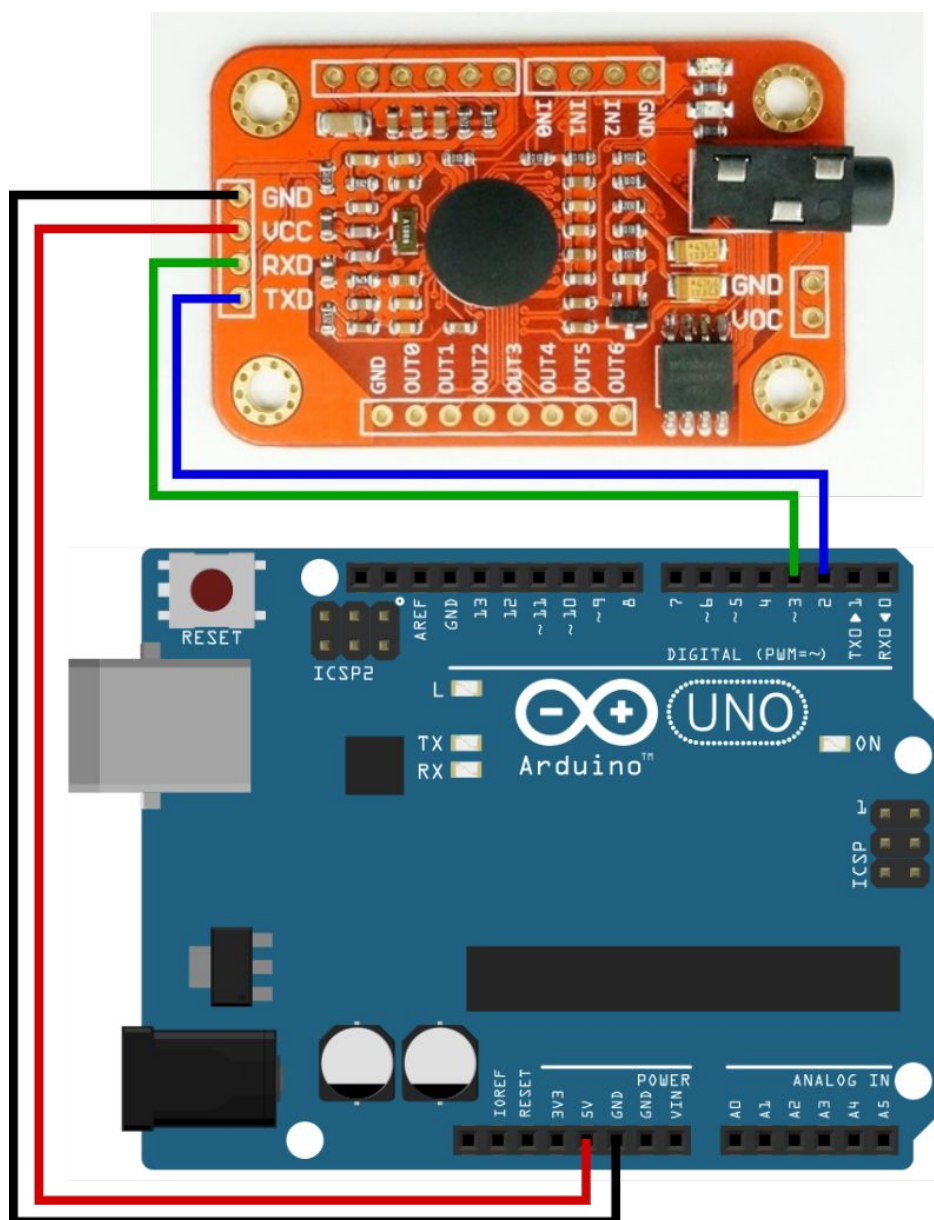


SPECIFIKACE

Pracovní napětí	4,5 až 5,5 V DC	Délka příkazu	1500 ms
Proud	<40 mA	Vstup pro mikrofon	AUX
Komunikační rozhraní	UART	Průměr mont. otvorů	3 mm
Napěťová logická úroveň	5 V DC	Rozteč mont. otvorů (mm)	39 x 22
Paměť	80 příkazů	Rozměry (mm)	50 x 30



ZAPOJENÍ



Pokud bude uživatel modul používat s Arduinem Mega 2560 připojí pin RX na pin 11 a pin TX na pin 12. Daná změna zapojení musí být upravena i v programu.

Aby bylo možné program zkompileovat, je nezbytné, aby uživatel stáhl knihovnu [VoiceRecognitionV3](#) a naimportoval ji do Arduino IDE.

Sériový terminál musí být nastaven na baudrate 115200 bps a nové řádkování (NL).

Aby byl uživatel schopen ovládat pomocí hlasových příkazů např. relé, je nutné nejprve hlasové příkazy přidat do modulu hlasových příkazů. K tomu slouží příklad knihovny *vr_sample_train.ino*. Díky tomuto příkladu přejde zařízení do učícího režimu a je schopné ukládat příkazy do své interní paměti.

Pro spínání relé poté stačí do Arduina nahrát příklad *vr_sample_relay.ino*. S tímto softwarem je uživatel schopen ovládat relé pomocí hlasových příkazů. Jak zapojit relé je vysvětleno v tomto [článku](#). Oba programy jsou volně ke stažení na stránce produktu v sekci *datasheety*.

Software je koncipován pro platformy Arduino Uno a Nano. Pokud bude uživatel modul používat s Arduino Mega 2560, připojí pin RX na pin 11 a pin TX na pin 12. Daná změna zapojení musí být upravena i v programu.

Rozšířené možnosti nastavení hlasového modulu nalezne uživatel v dokumentu *V3_manual.pdf*, který je rovněž volně ke stažení na stránce produktu.

Přidání hlasového příkazu

- uživatel nahraje do Arduina program *vr_sample_train.ino*
- poté otevře sériový terminál a nastaví jej (vizte výše)
- jakmile se načte menu programu, uživatel na vstup sériové linky napíše spojení např. *train 0*, což spustí učící režim pro příkaz s indexem 0 (indexů je celkem 0 až 79) – uživatel odešle příkaz
- jakmile se na výstupu objeví příkaz *Speak now*, uživatel vysloví příkaz
- jakmile se na výstupu objeví příkaz *Speak again*, uživatel příkaz z opakuje
- pokud bylo přidání příkazu úspěšné, program zareaguje zprávou *Success* – v případě, že program zareaguje zprávou *Can't match*, proces prosím zopakujte (příkazy se neshodovaly)
- takto postupujte u obou příkazů – příkaz s indexem 0 pro sepnutí relé, příkaz s indexem 1 pro rozepnutí relé
- jakmile uživatel příkazy úspěšně nahraje, musí je uložit do paměti zařízení – k tomu slouží příkaz *record 0* (příkaz musí opět následovat požadovaný index) – tuto akci provede uživatel pro oba příkazy
- pokud uložení proběhlo úspěšně, modul zareaguje zprávou *Trained*

```
train 0
-----
Record: 0      Speak again
Record: 0      Cann't matched
Record: 0      Speak now
Record: 0      Speak again
Record: 0      Success
Train success: 1
Record 0      Trained
-----
train 1
-----
Record: 1      Speak now
Record: 1      Speak again
Record: 1      Cann't matched
Record: 1      Speak now
Record: 1      Speak again
Record: 1      Success
Train success: 1
Record 1      Trained

Check 1 records.
1 record trained.
0      -->   Trained
-----
record 1
-----
Check 1 records.
1 record trained.
1      -->   Trained
```

Spínání relé pomocí hlasového příkazu

- uživatel připojí relé (spíná v log. 0) k Arduino - IO pin 5
- následně nahraje do Arduina program *vr_sample_relay.ino* – software byl navržen za předpokladu, že příkaz pro sepnutí relé je na indexu 0 a pro rozepnutí na indexu 1 (vizte přidání hlasového příkazu)
- uživatel vyslovuje nahrané příkazy a tím spíná relé