PiPeeper

Zastosowania systemów wbudowanych – projekt

Adrian Frydmański, Bohdan Jaminski

# Cel i założenia

* Stworzenie systemu do podglądu obrazu z kamery USB
* Dostęp przez stronę internetową
* Sterowanie kamerą – obracanie
* Użycie serwomechanizmów modelarskich

# Wykonanie

Do Raspberry Pi podłączony jest za pomocą magistrali I2C kontroler serw generujący sygnał PWM. Zasilany jest z zewnętrznego źródła zasilania, gdyż generowałby zbyt duże zakłócenia na linii 5V w Raspberry Pi. Kamerka internetowa podłączona jest za pomocą kabla USB.

Interfejs użytkownika został napisany w Pythonie i opiera się o framework Flask. Po wczytaniu strony wysyłany jest strumieniowo obraz z kamery. Ponadto po wciśnięciu przycisku w przeglądarce i otrzymaniu po stronie serwera odpowiedniego żądania zmieniane są wartości pozycji serwomechanizmów.

Komunikacja z serwerem odbywa się przy pomocy jQuery, które po wciśnięciu przycisku na stronie wysyła żądanie do serwera, skutkujące poruszeniem się serwomechanizmu (serwomechanizmów).



Rysunek PiPeeper z podłączonym powerbankiem do zasilania serw

# Testowanie i uruchamianie

1. Pobranie z repozytorium https://github.com/Adrian94F/PiPeeper
2. Włączenie obsługi I2C
3. Instalacja pipa (sudo apt install python-pip)
4. Instalacja Flaska (pip install flask)
5. Uruchomienie serwera poprzez main.py w katalogu głównym repozytorium
6. Uruchomienie strony http://adres\_raspberry\_pi:5000 (na Raspberry Pi localhost:5000)

# 

Rysunek 2 Interfejs użytkownika

# Wnioski

Projekt pokazał, jak można użyć komputera Raspberry Pi do własnych potrzeb i niskim nakładem pracy stworzyć zamiennik gotowego urządzenia, jakim jest typowa kamera IP. System może być rozbudowany o dodatkowe funkcje, na przykład nagrywanie obrazu, wykonywanie zdjęć, dodatkowe kamery, czy panel logowania i administrowania.