

Mgr inż. Stanisław GRZYWIŃSKI  
Inż. Artur OLENDER  
Inż. Mateusz WÓJTOWICZ  
Inż. Wiktor ZIĘBA  
Wojskowa Akademia Techniczna

DOI: 10.17814/mechanik.2015.7.239

## SYSTEM WSPOMAGANIA DOWODZENIA NA SZCZEBLU PODODDZIAŁU CZĘŚĆ 2. ROZWIĄZANIA PROGRAMOWE

*Streszczenie: W artykule przedstawiono rozwiązania programowe urządzenia mechatronicznego przeznaczonego do wspomagania dowodzenia w czasie działań bojowych żołnierzy na polu walki.*

### SUB-UNIT LEVEL COMMAND SUPPORT SYSTEM PART 2. SOFTWARE

*Abstract: The article presents software solutions for the mechatronic device designed to support command during combat troops on the battlefield.*

*Słowa kluczowe: Raspbbery Pi, IMU, GPS, stabilizacja obrazu*  
*Keywords: Raspbbery Pi, IMU, GPS, image sabilization*

## 1. WPROWADZENIE

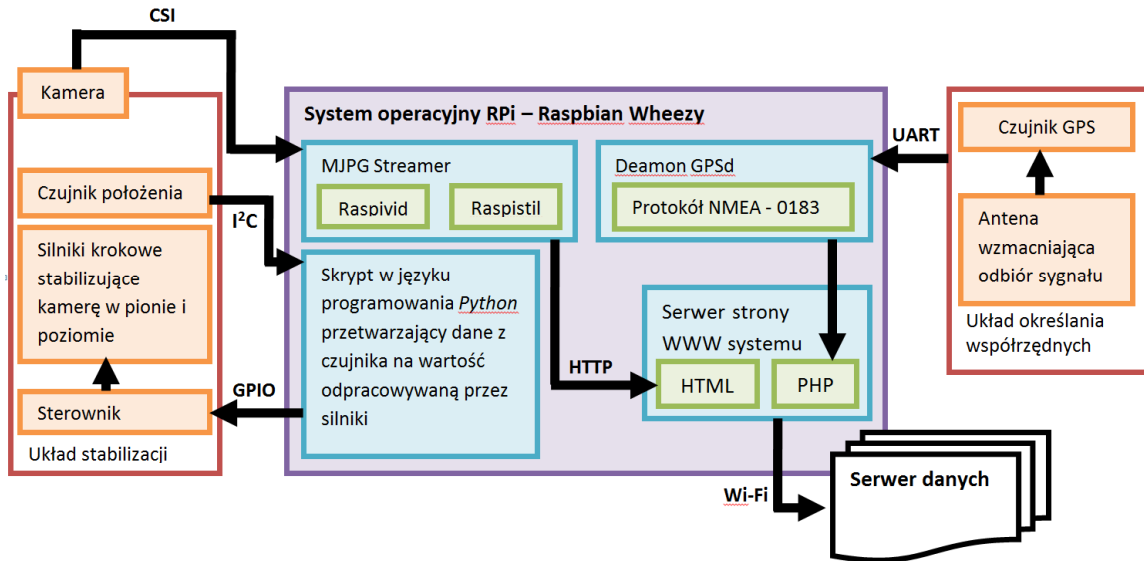
Rozwiązanie sprzętowe koncepcji systemu wspomagania dowodzenia opisane w artykule [1] do poprawnej pracy i wykorzystania w pełni swoich funkcjonalności potrzebuje odpowiedniego oprogramowania.

Celem niniejszego artykułu jest zaprezentowanie na przykładzie opracowanych rozwiązań programowych koncepcji systemu wspomagania dowodzenia na szczeblu małego pododdziału wielu problemów, jakie można napotkać, projektując oprogramowanie dla urządzeń wielosensorowych z cyfrową transmisją danych czasu rzeczywistego, działających w sieciach WLAN.

## 2. ROZWIĄZANIA PROGRAMOWE

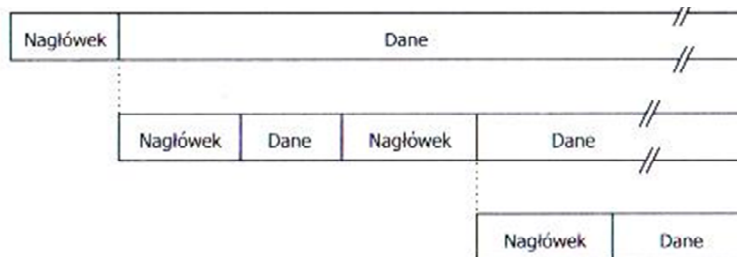
Schemat funkcjonalny oprogramowania systemu wspomagania dowodzenia na szczeblu pododdziału przedstawiono na rys. 1. Działanie systemu oparto na dedykowanym dla komputera Raspberry Pi systemie operacyjnym *Raspbian Wheezy*, który jest jedną z wielu dystrybucji *Linuxa*. Wybór umożliwił oprogramowanie urządzenia oraz pracę w trybach konsolowym przy wykorzystaniu powłoki *Bash* (ang. *Bourn Again Shell*) i graficznym – *GUI* (ang. *Graphical User Interface*). Oprogramowania z rodziny *Linux* umożliwiają edycję jądra systemu oraz jego plików źródłowych, przez co możliwe jest precyzyjne dostosowanie systemu operacyjnego do potrzeb sprzętowych [2]. Pliki systemowe wraz z opracowanym

oprogramowaniem zostały zapisane na karcie SD, przez co znacznie zmniejszono rozmiary urządzenia.



Rys. 1. Schemat funkcjonalny oprogramowania systemu wspomagania dowodzenia na szczeblu pododdziału

W celu uruchomienia karty sieciowej oraz utworzenia połączenia sieciowego WLAN konieczne było zaimplementowanie na urządzeniu sterownika karty sieciowej oraz wykorzystanie programu *HostAPD*. Program ten jest to demon przestrzeni użytkownika służący do obsługi bezprzewodowych punktów dostępu oraz uwierzytelniania serwerów. Umożliwia utworzenie działającego w tle punktu dostępu dla innych komputerów oraz jego zabezpieczenie w postaci hasła. W rozwiązaniach programowych wykorzystano również oprogramowanie *isc-dhcp-server*, które odpowiada za udostępnienie danych konfiguracyjnych, takich jak adres IP (ang. *Internet Protocol*) hosta, brama sieci, adres serwera DNS (ang. *Domain Name System*) i maska podsieci, łączącym się z RPi urządzeniem. Po skonfigurowaniu obu programów oraz edycji plików systemowych odpowiadających za rozruch systemu utworzono uruchamiającą się wraz z urządzeniem sieć WLAN. Konfiguracja karty i utworzenie sieci umożliwi elastyczne strumieniowanie danych. Struktura strumienia danych przedstawiona jest schematycznie na rys. 2.



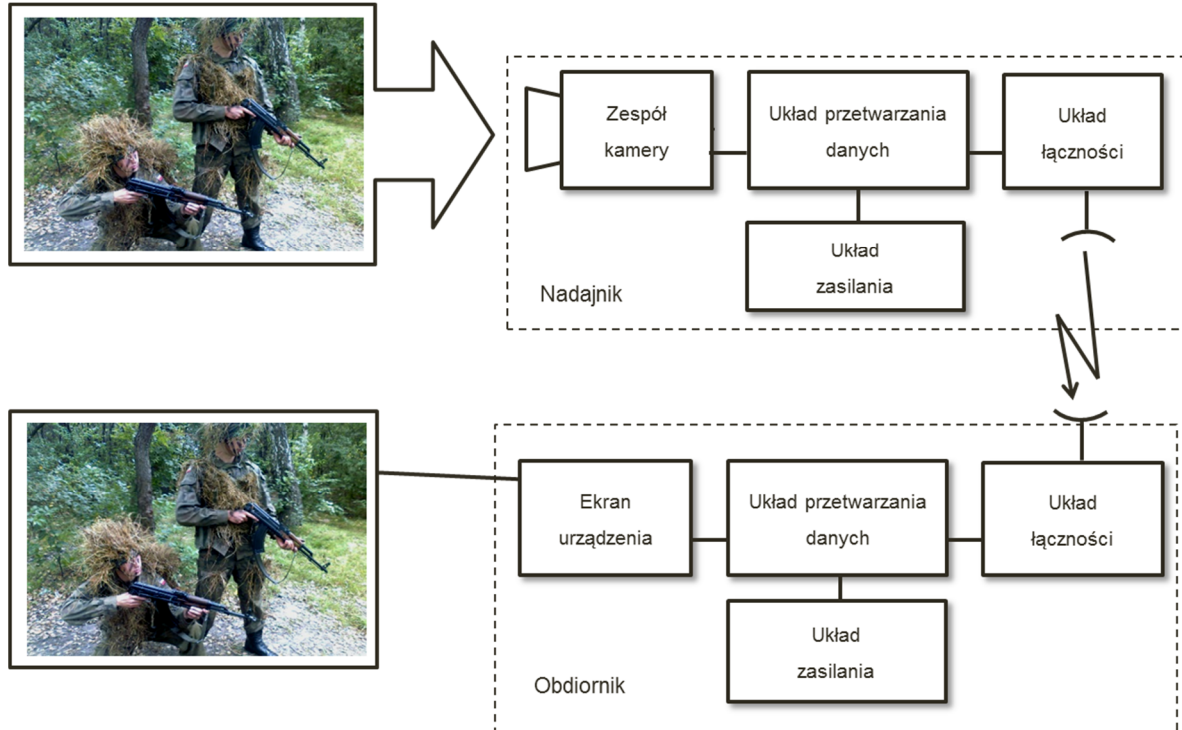
Rys. 2. Struktura strumienia danych [3]

Zadanie zobrazowania sytuacji taktycznej zrealizowano, wykorzystując dedykowaną dla RPi kamerę. Konfiguracja programowa sprowadzała się do zaadaptowania sterowników *Raspivid* i *Raspistill*. Po przeprowadzeniu testów różnych mediów, protokołów i metod strumieniowania danych multimedialnych, wybrany został najbardziej efektywny sposób opierający się na strumieniowaniu filmów w formacie MJPEG (ang. *Motion Joint Picture Expert Group*), czyli sekwencji następujących po sobie, niewiele różniących się obrazów [4].

Strumieniowanie wideo w formacie MJPEG rozdziela film na ciąg danych określających kolejno wszystkie informacje dotyczące przesyłanego obrazu. Obraz w postaci cyfrowej jest zbiorem macierzy, a strumień tego obrazu jest ciągiem następujących po sobie wartości danej macierzy. Wybrany do transmisji tego strumienia protokołem jest HTTP (ang. *HyperText Transfer Protocol*). Sesja HTTP odpowiadająca za transmisję danych jest sekwencją żądań i odpowiedzi sieciowych, przy czym żądania wysyłane są przez użytkownika (moduł master), a odpowiedzi przez serwer posiadający dostęp do plików źródłowych (moduł slave). Dzięki wykorzystaniu tego protokołu odbiór danych możliwy będzie na praktycznie każdym urządzeniu, ponieważ jest on standardowym protokołem wykorzystywanym przez przeglądarki internetowe. Za tworzenie obrazów oraz ich strumieniowanie przez sieć odpowiada program MJPG Streamer. Tworzone przez niego obrazy przesyłane są do mikrokomputera interfejsem CSI. Oprogramowanie RPi przetwarza obraz na strumień danych i wysyła go przy wykorzystaniu interfejsu Wi-Fi do modułu master. Dzięki wykorzystaniu formatu MJPEG nie są przesyłane wszystkie dane dotyczące poszczególnych klatek podglądu, a tylko te, które uległy zmianie względem poprzedniej klatki. Rozwiązanie to zwiększa płynność transmitowanego obrazu i zapewnia efektywniejsze wykorzystanie procesora RPi. Do zapewnienia poprawnego działania aplikacji konieczna była implementacja bibliotek, takich jak:

- *Libjpeg62-dev*, która usprawnia obsługę multimediów opartych na formacie JPEG [5],
- *Imagemagick*, umożliwiająca tworzenie, edycję, kompresję i konwersję obrazów [6].

Dzięki wbudowanemu modułowi sieciowemu możliwe jest umieszczenie danych na stronie internetowej o adresie będącym numerem IP mikrokomputera RPi wraz z odpowiednim portem sieciowym. Schemat funkcjonalny układu transmisji podglądu przedstawiono na rys. 3.



Rys. 3. Schemat funkcjonalny układu transmisji podglądu

Za stabilizację kamery odpowiada podsystem zbudowany przy wykorzystaniu żyroskopu i akcelerometru czujnika MPU-6050 oraz dwóch silników krokowych wraz ze sterownikami.

Układ pomiarowy wykorzystuje do transmisji danych magistralę I<sup>2</sup>C, natomiast sterowanie silnikiem możliwe jest poprzez wykorzystanie interfejsu ogólnego przeznaczenia GPIO.

W celu zapewnienia poprawnej komunikacji czujników pomiarowych oraz urządzeń wykonawczych z procesorem, niezbędne było zainstalowanie sterowników do obsługi interfejsu GPIO oraz I<sup>2</sup>C. W celu realizacji zadania stabilizacji kamery opracowano skrypt, który został napisany w języku *Python* z wykorzystaniem środowiska *IDLE*. W programie zostały zdefiniowane funkcje inicjujące transmisję danych oraz określające parametry pracy czujnika IMU. Po każdorazowym uruchomieniu programu następuje także automatyczna kalibracja przyrządów pomiarowych. Układ stabilizacji działa z wykorzystaniem pętli sprzężenia zwrotnego. Sygnałem zadanym jest wartość kąta w płaszczyźnie x oraz y. Sygnałem z pętli sprzężenia zwrotnego jest bieżąca pozycja kamery mierzona przez czujnik MPU-6050. Na podstawie uchybu wyliczany jest sygnał sterujący podawany na sterownik silnika krokowego, a następnie na silnik. Po zainicjowaniu wszystkich procedur startowych systemu *Raspbian Wheezy* następuje automatyczne uruchomienie się skryptu stabilizującego kamerę.

W celu określania położenia żołnierza zdecydowano się wykorzystać odbiornik *Adafruit Ultimate GPS* [8]. Sygnał odebrany z satelitów jest przesyłany przez magistralę UART do mikrokomputera RPi. Dzięki wcześniejszej instalacji *daemon* *gpsd* wraz z programem *Foxtrot GPS* możliwa jest obróbka tych danych oraz zobrazowanie aktualnej pozycji żołnierza na specjalnie w tym celu przygotowanej stronie internetowej (rys. 4).

Projekt koncepcyjny systemu lokalizacji położenia żołnierza na polu walki - GPSD Test Station 52.251748333, 20.897453333 - Moduła Pi40

Projekt koncepcyjny sy...

192.168.137.109/gps/gpsd.php

Często odwiedzane - Getting Started - programy naukowe - gpszobrazowanie - cern - fox - serwer global - howto host - yadda - wesolych swiat - zaklady

Server GPS został umieszczony na platformie Raspberry Pi.  
Projekt strony został opracowany na potrzeby pracy inżynierskiej przez kpr.pchor Wiktor ZIEBA

Bieżące informacje	
Czas (UTC)	2014-12-15T20:51:25.000Z
Szerokość	52.251748333
Długość	20.897453333
Wysokość	135.6
Fix Type	3
Satellity	14
HDOP	1.26
VDOP	1.82

DOP OPIS	
1	idealny
2-3	znakomity
4-6	dobry
7-8	umiarkowany
9-20	słaby
> 20	zły

Bieżące satelity				
PRN	Elewacja	Azymut	SS	Używana
15	60	269	17	T
26	55	169	25	T
28	48	59	17	T
17	38	121	16	N
13	32	153	21	T
24	31	262	0	N
18	20	305	15	T
30	12	94	13	T
11	0	41	0	N
22	5	357	0	N
4	4	30	14	N
12	4	227	0	N
1	3	58	0	N
46	0	0	0	N

This script is distributed by the [GPSD project](#).

Rys. 4. Widok mapki zobrazującej położenie żołnierza

Jako medium udostępniania i wizualizacji danych wykorzystano, podobnie jak w przypadku kamery, technologię WWW (*World Wide Web*) oraz język programowania HTML5. Dzięki nowym komponentom języka, takim jak *Canvas*, możliwe było utworzenie szybko odświeżających się elementów, dzięki którym film w postaci następujących po sobie obrazów w formacie JPEG mógł być edytowany i wyświetlany na stronie (rys. 5). Strona internetowa powstała w środowisku Notepad ++ 6.7.5.0 PL dała możliwość prezentacji współrzędnych żołnierza w czasie rzeczywistym, wyświetlania, edycji podglądu oraz zapisu zdjęć wykonanych przez kamerę modułu slave.

## SOST

O projekcie

Podgląd

Edycja

Zdjęcie

## Zdjęcie

Wybranie zakładki "Zdjęcie" powoduje pobranie jednej klatki podglądu (następującej w momencie wybrania zakładki) oraz umożliwia jego edycję poprzez wybranie jednego z utworzonych filtrów.



Obraz oryginalny



Obraz edytowany

Desaturacja obrazu	Inwersja obrazu	Rozjasnienie obrazu
Sciemnienie obrazu	Wyostrez obraz	Reset

Rys. 5. Widok strony internetowej wyświetlającej jedną klatkę z kamery zobrazowania sytuacji taktycznej

Fragment kodu programu odpowiadającego za wyświetlanie filmu oraz jego edycję:

```
[...]
<figure>
  <canvas id="imgCanvas" width="800" height="600" style="border:1px solid
  #d3d3d3;"/></canvas>
  <figcaption>Podgląd obrazu</figcaption>
</figure>
<figure>
  <canvas id="myCanvas" width="800" height="600" style="border:1px solid
  #d3d3d3;"></canvas>
  <figcaption>Edytowany obrazu</figcaption>
</figure>
<div id="txt" style="display: none;">0</div>
<script type="text/javascript">
var iUpdateMs = 10;
  var count = 0;
  var newImg;

  setTimeout("imageupdate()", iUpdateMs );
  function imageupdate(){
    document.getElementById("txt").innerHTML = count++;
    newImg = new Image();
    newImg.Id = "camimg" + count;
    newImg.Name = newImg.Id;
    newImg.onload = imageLoaded;
    newImg.src = "?action=snapshot&n=" + count;
  }
  function imageLoaded(){
    var drawingCanvas = document.getElementById("imgCanvas");
    if(drawingCanvas.getContext){
      var context = drawingCanvas.getContext("2d");
      context.drawImage(newImg,0,0,800,600,0,0,800,600)
    }
    setTimeout("imageupdate()", iUpdateMs);
  }
</script>
<button onclick="desaturacja()">Desaturacja obrazu</button>
<script>
var interwal;
function desaturacja() {
clearInterval(interwal);
var FPS = 30;
interwal = setInterval(function() {
  draw();
}, 1000/FPS);
function draw() {
```

```

var c = document.getElementById("myCanvas");
var ctx = c.getContext("2d");
var img = document.getElementById("imgCanvas");
ctx.drawImage(img, 0, 0);
var imgData = ctx.getImageData(0, 0, c.width, c.height);
var i;
for (i = 0; i < imgData.data.length; i += 4) {
    var brightness = 0.34 * imgData.data[i] + 0.5 * imgData.data[i + 1] + 0.16 *
imgData.data[i + 2];
    imgData.data[i] = brightness;
    imgData.data[i+1] = brightness;
    imgData.data[i+2] = brightness;
    imgData.data[i+3] = 255;
}
ctx.putImageData(imgData, 0, 0);
}
}
</script>
[...]
```

### 3. PODSUMOWANIE

Podjęte prace dotyczące modelu systemu wspierającego dowodzenie na szczeblu małego pododdziału, którego zadaniem jest określenie położenia żołnierza w terenie oraz zobrazowanie sytuacji taktycznej, mogą stanowić podstawę do opracowania, innowacyjnego w technice wojskowej, systemu obserwacji sytuacji taktycznej. System wykorzystany w pododdziałach plutonu, czy drużyny nie został jeszcze opracowany i wyprodukowany. Sprawia to, że koncepcja wykorzystania takiego urządzenia jest innowacyjna w dziedzinie techniki wojskowej.

Problem określania położenia żołnierza w terenie oraz zobrazowania sytuacji taktycznej rozwiązany został w ramach realizacji projektu studenckiego RA 2014-2015. Opracowane rozwiązania sprzętowo-programowe wymagają wiele pracy.

Ze względu na równoległe tworzenie modułów aplikacji, przy tworzeniu algorytmów wykorzystano system kontroli wersji *Perforce* oraz *Subversion* [9], które pozwoliły na łatwe zarządzanie kodem źródłowym projektowanego rozwiązania.

Dzięki wykorzystaniu protokołu HTTP oraz języka programowania HTML5 będącego nowym standardem stron WWW (ang. *World Wide Web*) odbiór danych transmitowanych strumieniowo z modułu slave wspomagania dowodzenia na szczeblu pododdziału możliwy jest na każdym urządzeniu posiadającym kartę sieciową obsługującą transmisję Wi-Fi oraz przeglądarkę internetową kompatybilną z językiem HTML5.

System pozwala na odbiór płynnego podglądu, nie posiadającego opóźnień, które umożliwi rozpoznanie sylwetki człowieka znajdującego się w odległości do około 60 m od miejsca, w którym znajduje się model urządzenia. Ponadto pozwala zwizualizować współrzędne żołnierza w czasie rzeczywistym, dzięki czemu zrealizowano zakładaną funkcjonalność urządzenia. W celu podniesienia wartości realizowanego projektu na stronie

zostały zaimplementowane skrypty napisane w języku JavaScript umożliwiające nałożenie filtrów na obraz wideo, które mogą ułatwić wzrokową obserwację dowódcy prowadzoną na ekranie odbiornika. Strona posiada funkcję wykonywania zdjęcia oraz zapisania go na dysku odbiornika. Zapisane zdjęcia wraz ze współrzędnymi geograficznymi mogą posłużyć dowódcy pododdziału do wykonania meldunku z przeprowadzanej operacji bądź w celach szkoleniowych.

\*\*\*

*Praca finansowana ze środków Prorektora ds. studenckich WAT  
jako projekt studencki realizowany w RA 2014-2015.*

## LITERATURA

- [1] Grzywiński S., Olender A., Wójtowicz M., Zięba W.: *System wspomagania dowodzenia na szczeblu pododdziału. Cz. 1. Rozwiązanie sprzętowe*, materiały konferencyjne, Tom 1, XIX Międzynarodowa Szkoła Komputerowego Wspomagania Projektowania, Wytwarzania i Eksploatacji, 11-15 maja, Jurata 2015.
- [2] Welsh M., Dalheimer M., Kaufman L.: *Linux*, RM, Warszawa 2000.
- [3] Antosik B.: *Transmisja internetowa danych multimedialnych w czasie rzeczywistym*, Wydawnictwo Komunikacji i Łączności, Warszawa 2012.
- [4] Zieliński T.: *Cyfrowe przetwarzanie sygnałów. Od teorii do zastosowań*, Wydawnictwo Komunikacji i Łączności, Warszawa 2009.
- [5] <http://www.imagemagick.org/>
- [6] <https://subversion.apache.org/>
- [7] <http://www.cmake.org/>
- [8] Kaniewski P.: *System nawigacji satelitarnej GPS. Część 7. Pozycja, prędkość i czas*: Elektronika praktyczna, nr 8/2006, Warszawa, s. 97-99.
- [9] Grzywiński S.: *Perforce – system kontroli wersji w zastosowaniach*, Mechanik, nr 7/ 2014.