

Mateusz Kubaszek\*  
Jan Macheta\*  
Łukasz Krzak\*  
Cezary Worek\*  
Szymon Zawora\*\*

\*AGH Akademia Górniczo-Hutnicza  
Wydział Informatyki, Elektroniki i Telekomunikacji, Katedra Elektroniki  
al. Adama Mickiewicza 30, 30-059 Kraków

\*\*Zamel Sp. z o.o.,  
ul. Zielona 27, 43-200 Pszczyna  
kubaszek@agh.edu.pl

**KSTiT 2019**

Wrocław, 26-28 czerwca 2019

## MODEL ZUŻYCIA ENERGII DLA WĘZŁÓW SIECI 6TiSCH W PAŚMIE SUB-GHZ

### ENERGY CONSUMPTION MODEL FOR 6TiSCH NETWORK NODES IN SUB-GHZ BAND

Streszczenie: Standard IEEE 802.15.4-TSCH (ang. Time Slotted Channel Hopping) definiuje warstwę fizyczną oraz warstwę łącza dla sieci LLN (ang. Low Power and Lossy Network) dopasowaną do rozwiązań IIoT (ang. Industrial Internet of Things). Powstający na jego bazie standard 6TiSCH (ang. IPv6 over TSCH), integrując się z klasycznym stosem internetowym, staje się funkcjonalnym rozwiązaniem dla przemysłowych sieci sensorowych. W artykule zaprezentowano model opisujący zużycie energii węzłów w sieci 6TiSCH, zaadaptowanej do pracy w paśmie 863-870 MHz. Model opiera się na danych pomiarowych pochodzących z implementacji stosu na autorskich modułach radiowych.

Abstract: IEEE 802.15.4-TSCH (Time Slotted Channel Hopping) standard defines the physical and link layer for LLNs (Low power and Lossy Network) suitable for IIoT (Industrial Internet of Things) solutions. The 6TiSCH (IPv6 over TSCH) standard created on its basis, that integrates with a classical Internet stack, becomes a functional solution for industrial wireless sensor networks. This article presents an energy consumption model for the 6TiSCH network nodes, operating in the 863-870 MHz band, based on measurements taken with an implementation of the stack on custom radio modules.

Słowa kluczowe: 6TiSCH, IIoT, sieci kratowe, RPL, 6lowpan, 6top, 6P, modelowanie poboru mocy, OpenWSN.

Keywords: 6TiSCH, IIoT, mesh networks, RPL, 6lowpan, 6top, 6P, energy consumption modelling, OpenWSN.

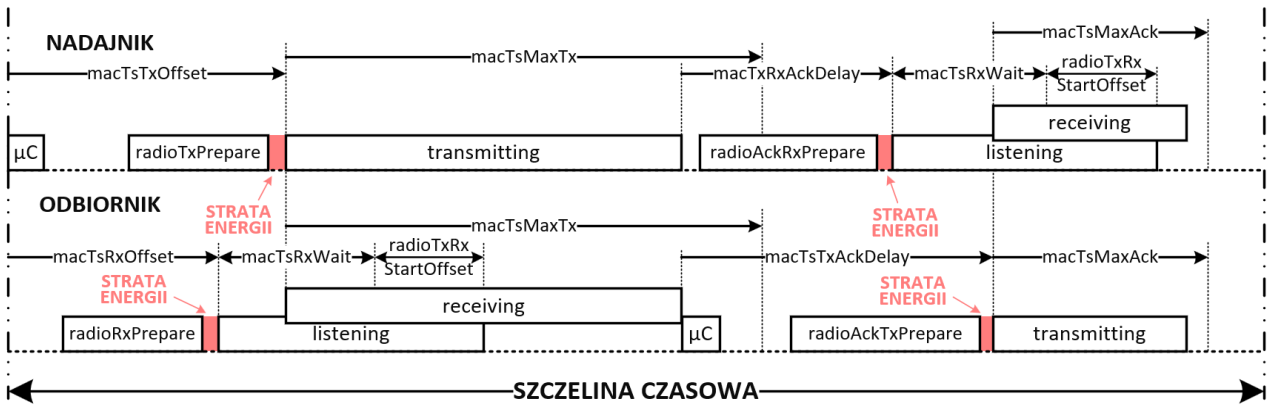
#### 1. WSTĘP

Sieci LLN pozwalają na połączenie dużej liczby urządzeń o ograniczonych zasobach sprzętowych w kratową sieć bezprzewodową. Głównym elementem umożliwiającym komunikację jest stos sieciowy dostarczający odpowiednią jakość usług (QoS, ang. Quality of Service). Od aplikacji IIoT wymaga się zapewnienia wysokiej niezawodności, wieloletniej pracy na baterii i dobrej integracji z Internetem przy zachowaniu niskiego kosztu urządzeń, co sprzyja poszukiwaniu i wdrażaniu nowych rozwiązań w zakresie radiowej transmisji danych.

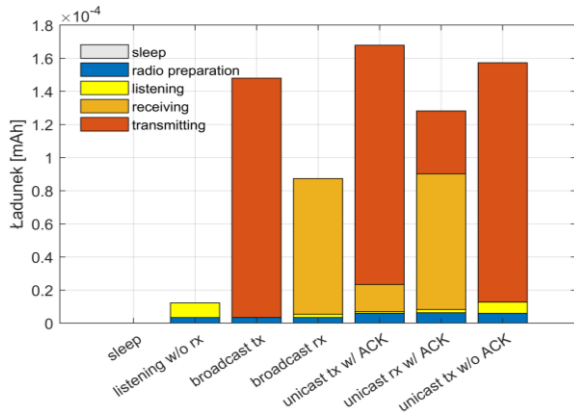
Technika TSCH zdobyła uznanie w zastosowaniach przemysłowych odpowiadając wymaganiom stawianym urządzeniom pracującym w takim środowisku [1]. Sieci pracujące w trybie TSCH pracują synchronicznie w wielu

kanałach częstotliwości, posługując się szczelinami czasowymi, które są alokowane w celu realizacji transmisji poszczególnych pakietów w sieci w obrębie cyklicznej superramki. Jeden z pierwszych protokołów pracujący w tym trybie, TSMP (ang. Time Synchronized Mesh Protocol), dał zadowalający rezultat w projektach pilotażowych wykazując się wysokim poziomem niezawodności, wyrażonym jako współczynnik dostarczania pakietów, który wyniósł ponad 99,9% [2]. Technika TSCH stała się również podstawą takich standardów przemysłowych jak Wireless-HART oraz ISA100.11a. W 2015 roku tryb TSCH został wprowadzony do standardu IEEE 802.15.4, definiując warstwę PHY oraz MAC, jednak bez rozwiązania kwestii sposobu alokacji szczelin czasowo-częstotliwościowych. Rozwiązania tej kwestii oraz integracji standardu IEEE 802.15.4-TSCH ze stosem 6LoWPAN podjęła się grupa robocza IETF 6TiSCH. W wyniku jej prac zaproponowano tzw. konfigurację minimalną, definiującą m.in. sposób zestawiania i podtrzymania działania sieci oraz interfejs planisty rezerwujący zasoby łącza (6P/6top).

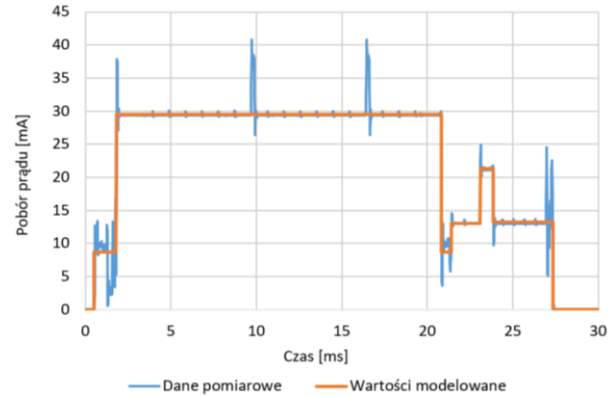
Standard 6TiSCH jest obecnie aktywnie rozwijany przez środowisko akademickie i komercyjne [3][4]. W większości projektów bazuje on na warstwie fizycznej pracującej w paśmie 2.4GHz z przepływnością 250kbps i modulacją DSSS-OQPSK. Tymczasem w aplikacjach IIoT, gdzie ważny jest zasięg transmisji radiowej, penetracja przeszkód, niższe tłumienie i mniejsze prawdopodobieństwo kolizji z innymi systemami, znacznie lepszym rozwiązaniem jest wykorzystanie pasm sub-GHz [5]. W niniejszym artykule zaprezentowano analizę zużycia energii zasilającej przez węzły pracujące w sieci 6TiSCH, zaadaptowanej do pracy w paśmie 863-870 MHz. Wynikowy model symulacyjny bazuje na danych eksperymentalnych z sieci czujników opartych o autorskie moduły radiowe, pracujące zgodnie z warstwą fizyczną IEEE 802.15.4 SUN PHY OM#1, wykorzystując 69 kanałów częstotliwościowych o szerokości 100 kHz oraz modulację FSK z przepływnością 50kbps. Uzyskane wyniki można jednak odnieść do innej, niestandardowej warstwy fizycznej lub platformy sprzętowej kompatybilnej ze stosem 6TiSCH lub ekstrapolować na rozwiązania o podobnych parametrach związanych z ruchem sieciowym.



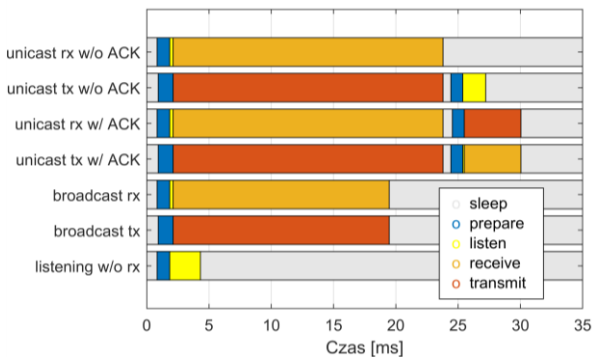
Rys. 1. Zależności czasowe występujące w obrębie szczeliny.



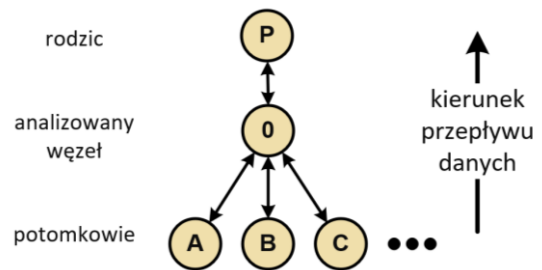
Rys. 2. Pobór ładunku ze źródła 3,3V dla poszczególnych typów szczelin.



Rys. 3. Profil poboru prądu modułu radiowego.



Rys. 4. Zależności czasowe występujące w różnych typach szczelin analizowanych w modelu.



Rys. 5. Przyjęty model topologii sieci.

## 2. MODEL ENERGETYCZNY

Głównym celem było opracowanie modelu symulacyjnego, umożliwiającego oszacowanie zużycia energii pobieranej ze źródła zasilania przez węzeł sieci 6TiSCH. Poniżej przedstawiono przyjęte założenia oraz dane wejściowe.

### 2.1. Szczelina czasowa

Podstawową jednostką zasobów łącza w standardzie IEEE 802.15.4-TSCH jest szczelina czasowa. W każdej szczelinie dwa węzły mogą wymienić pakiet danych wraz z potwierdzeniem. W szczelinie występuje kilka przedziałów czasowych opisanych przez parametry czasowe, różne dla szczelin nadawczych i odbiorczych (Rys.1). Parametry te stanowią zestaw atrybutów warstwy MAC i muszą uwzględniać możliwości sprzętowe zastosowanych układów radiowych i przetwarzających (MCU,

ang. Microcontroller Unit). Dla opracowanych modułów radiowych pracujących w paśmie sub-GHz wartości tych atrybutów zebrano w tabeli 1.

Tab. 1. Atrybuty MAC PIB

Parametr	Przyjęta wartość [ $\mu$ s]
macTsTxOffset	2120
macTsRxOffset	1560
macTsMaxTx	30000
macTsMaxAck	15000
macTsRxWait	1120
macTsAckWait	500
macTsTxAckDelay	2000
macTsRxAckDelay	1750

Dodatkowe parametry związane z czasem wykonywania się procedur oraz z szybkością działania układu ra-

diowego przedstawiono w tabeli 2. Zostały one wyznaczone empirycznie i są kompromisem gwarantującym skrócenie szczeliny przy jednoczesnym zachowaniu powtarzalności działania i minimalizacji opóźnień.

Tab. 2. Implementacyjne parametry czasowe szczeliny

Parametr	Przyjęta wartość [ $\mu$ s]
radioTxRxStartOffset	1350
radioTxPrepare	1200
radioRxPrepare	1000
radioAckTxPrepare	1050
radioAckRxPrepare	950

Parametr radioTxRxStartOffset jest definiowany jako różnica czasu pomiędzy momentem rozpoczęcia nadawania pakietu, a momentem rozpoznania słowa synchronizacyjnego w odbiorniku. Wydłużenie czasu nasłuchu o radioTxRxStartOffset jest zatem konieczne, aby nie przerwać przedwcześnie już rozpoczętego odbioru ramki. Pozostałe parametry obejmują przygotowanie danych przez MCU oraz skonfigurowanie układu radiowego tak, aby był w stanie gotowości do nadawania lub odbioru ramki w żądanym momencie trwania szczeliny czasowej.

## 2.2. Typy szczelin czasowych

Analizując sposób wykorzystania pojedynczej szczeliny czasowej możemy wyróżnić 8 rodzajów aktywności. Najprostszym przypadkiem jest szczelina, w której moduł pozostaje w stanie uśpienia. Pozostałe związane są z nasłuchem, nadawaniem lub odbiorem ramki. Zostały one przedstawione na rysunku 4 w postaci diagramów czasowych zakładając, że rozmiar danych w warstwie MAC dla transmisji typu *broadcast* wynosi 100B, a dla typu *unicast* 127B. Wśród diagramów, pierwszy od dołu ilustruje nasłuch bez odbioru, w czasie którego układ radiowy nasłuchuje w szczelnie nie rozpoczynając odbioru ramki. Ruch typu *broadcast*, przedstawiony jako kolejne dwa wykresy, nie wymaga potwierdzenia. Węzeł wraca do trybu uśpienia od razu po nadaniu lub odbiorze ramki. Kolejne cztery to różne wariacje ruchu typu *unicast* z potwierdzeniami. Na rys. 2 przedstawiono ładunek pobierany przez moduł dla każdego z omawianych typów szczelin czasowych z podziałem na składowe odpowiadające różnym aktywnościom układu radiowego w czasie trwania szczeliny.

## 2.3. Topologia sieci

Zgodnie z zaimplementowanym algorytmem trasowania (RPL ROLL), posługując się szeregiem reguł (m.in. DAG Rank) węzeł wybiera rodzica, do którego kieruje ruch w górę sieci. Do już podłączonego węzła dołączają się kolejne węzły, stając się jego potomkami. Generowane przez nie dane muszą zostać przesłane za jego pośrednictwem w górę sieci. Pakiety nie mogą być łączone, ponadto analizowany węzeł również uczestniczy w generowaniu danych. Rezerwowane są tylko pojedyncze dwukierunkowe szczeliny czasowe (w przypadku braku aktywności, oba węzły prowadzą nasłuch). Wynika z tego, że każdy węzeł posiada  $2+N$  aktywnych szczelin (szczelina rozgłaszająca, szczelina do rodzica i po jednej szczelinie do  $N$  potomków).

## 2.4. Konfiguracja stosu i model ruchu

Założono, iż każda superramka składa się z 53 szczelin czasowych, każda o czasie trwania 35 ms, co daje

okres obiegu wynoszący 1,855 s. W sieci istnienie ruch utrzymujący synchronizację, trasowanie oraz umożliwiający asocjacje nowych węzłów. Wiąże się z nim okresowe wysyłanie pakietów Beacon przez warstwę MAC oraz DIO (ang. DODAG Information Object) przez warstwę RPL. Pakiety warstwy 6P/6top, będące jednorazową lub rzadką transakcją, nie zostały uwzględnione.

Tab. 3. Podstawowe parametry konfiguracji stosu.

Liczba szczelin w superramce	53
Czas trwania szczeliny	35 ms
Okres superramki	1 855 ms
Okres wysyłania p. Beacon	Co 6-tą superramkę
Okres wysyłania p. DIO	Co 6-tą superramkę
Okres wysyłania danych aplikacyjnych	10 – 100 sekund
Rozmiar danych MAC pakietu aplikacyjnego	127 bajtów
Liczba potomków	1 - 10
Liczba aktywnych szczelin	(2 + l. potomków)

## 3. PLATFORMA SPRZĘTOWA ORAZ IMPLEMENTACJA STOSU

Implementacja prezentowanego stosu sieciowego 6TiSCH opiera się na projekcie OpenWSN rozwijanego w Uniwersytecie w Berkeley [6]. Oprogramowanie to wykorzystuje architekturę sterowaną zdarzeniami (ang. event-driven). Krytyczne czasowo funkcje stosu działają w przerwaniami sprzętowych. Reszta operacji wykonywanych jest w zakolejkowanych zadaniach w trybie nieuprzywilejowanym bez wywłaszczania. Stos uruchomiono na autorskiej platformie sprzętowej w postaci modułu radiowego opartego na układzie EZR32WG330F256R67G. Integruje on wysokiej klasy układ radiowy pracujący w zakresie sub-GHz z rdzeniem ARM Cortex-M4 oraz użytecznymi układami peryferyjnymi. Podstawowe parametry modułu zebrano w tabeli 4. Uruchomienie nowej platformy sprzętowej wiązało się z implementacją sterownika radiowego kompatybilnego z SUN PHY OM #1 oraz implementacją niskopoziomowego interfejsu stosu.

Tab. 4. Podstawowe parametry modułu radiowego

Graniczna czułość (15% PER)	-100 dBm
Moc nadawcza	+13 dBm
Pasmo pracy	863 – 870 MHz
Napięcie zasilania	3,3 V
Prąd w trybie uśpienia	od 1 $\mu$ A do 2 $\mu$ A
Klasa wg. ETSI 300 220	Klasa 1.5
Szybkość wybudzenia MCU	2 $\mu$ s
Dokładność zegara	$\pm$ 10ppm
Interfejsy I/O (w kilku konfiguracjach łącznie/rozłącznie)	GPIO, USART, SPI, I2C, ADC, DAC, Timer
Wymiary modułu	18 mm x 20 mm

Wyniki pomiaru profilu poboru prądu autorskiego modułu radiowego zostały przedstawione w postaci graficznej na rysunku 3. Prezentuje on również przyjęte w modelu charakterystyczne przedziały czasu pracy układu radiowego i uśredniony dla nich prąd węzła, którego wartości zostały zamieszczone w tabeli 5.

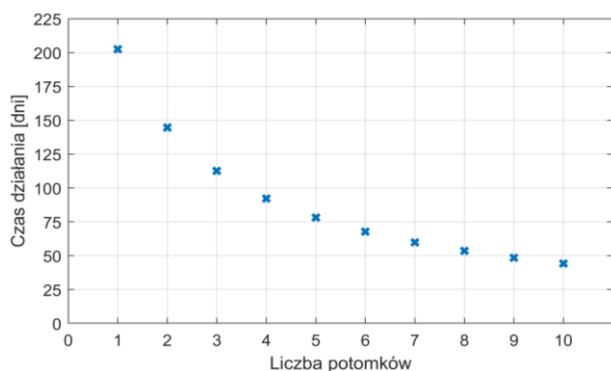
## 5. WNIOSKI

Tab. 5. Pobór prądu modułu radiowego

Parametr	Pobierany prąd [mA]
radioSleepCurrent	0,002
radioSynthCurrent	8,69
radioListeningCurrent	13,07
radioReceivingCurrent	13,23
radioStartReceiving	21,34
radioTransmitCurrent	29,56

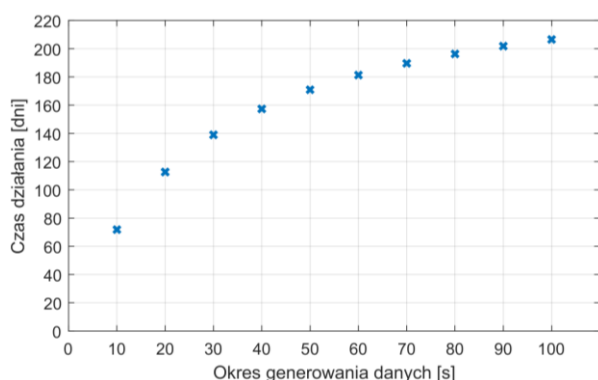
## 4. WYNIKI SYMULACJI

Przykładowe wyniki symulacji, bazującej na opisanych wyżej założeniach zostały przedstawione na wykresach. Na rys. 6 przedstawiono liczbę dni pracy modułu radiowego ze źródłem zasilania o pojemności 1Ah. Założono, że pakiet z danymi jest generowany w każdym węźle co 20 sekund. Badany węzeł trasuje wszystkie pakiety z danymi do swojego rodzica (rys. 5).



Rys. 6. Czas działania węzła zasilanego ze źródła o pojemności 1Ah dla okresu wysyłania danych wynoszącego 20 sekund.

Na rys. 7 przedstawiona została zależność czasu pracy węzła sieci zasilanego ze źródła o pojemności 1Ah, w funkcji długości okresu wysyłania danych aplikacyjnych. Dąży ona asymptotycznie do wartości ok. 260 dni, kiedy w sieci istnieje tylko ruch utrzymaniowy oraz nasłuch w aktywnych szczelinach.



Rys. 7. Czas działania węzła sieci zasilanego ze źródła o pojemności 1Ah trasującego ruch swój i 3 potomków.

Uzyskane wyniki wykazują silny związek liczby transmisji wykonywanych przez węzły sieci z poborem energii. Wyniki symulacji dowodzą (rys. 2), że szczeliny, w których prowadzony jest nasłuch pobierają o rząd wielkości mniej energii niż szczeliny przeznaczone do nadawania, co jest bardzo pożądanym rezultatem.

Opracowany model pozwolił na oszacowanie czasu pracy węzła dla danych scenariuszy użycia i konfiguracji warstw MAC oraz PHY. Z przeprowadzonych symulacji wynika, że w sieci krytyczne są kolizje pakietów w szczelinach współdzielonych i w szczelinach komunikacji dwustronnej. W sieciach, w których dostępny jest duży zapas szczelin komunikacyjnych opłacalne jest rezerwowanie szczelin jednokierunkowych. Ich wykorzystanie nie powoduje energochłonnych retransmisji, będących wynikiem kolizji pakietów. Istotną cechą jest również równoważenie ruchu w sieci pomiędzy wszystkimi węzłami. Sieć charakteryzuje się dużym wpływem liczby sąsiadów na czas pracy węzła. Protokół trasowania powinien premiować węzły o małej liczbie połączeń przy wyszukiwaniu nowych ścieżek, redukując w ten sposób liczbę węzłów pośredniczących w ruchu w górę sieci.

Praca powstała w ramach projektu „Innowacyjny system niskoenergetycznej, radiowej sieci sensorowej dedykowanej dla środowisk przemysłowych i komercyjnych, wspierający usługi monitorowania procesów produkcyjnych i optymalizacji zużycia mediów”, współfinansowanego w ramach Regionalnego Programu Operacyjnego Województwa Śląskiego na lata 2014-2020, oś priorytetowa 1. Nowoczesna gospodarka, Działanie 1.2. Badania, rozwój i innowacje w przedsiębiorstwach, nr umowy UDA-RPSL.01.02.00-24-0664/16-00.

## LITERATURA

- [1] Bello Philip. 1963. „Characterization of randomly time-variant channels”. *IEEE Transactions on Communications Systems*, 11 (4) : 360–393.
- [2] Kristofer Pister, Pance Doherty. 2008. „TSMP: Time Synchronized Mesh Protocol”. *Proceedings of the IASTED International Symposium, Distributed Sensor Networks*, pp. 391-398.
- [3] Pascal Thubert, Thomas Watteyne, Maria Rita Palatella, Xavier Vilajosana, Qin Wang. 2013. “IETF 6TSCH: Combining IPv6 Connectivity with Industrial Performance”. *2013 Seventh International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, pp. 542-546.
- [4] Maria Rita Palatella, Nicola Accettura, Xavier Vilajosana, Thomas Watteyne, Luigi Alfredo Grieco, Gennaro Boggia, Mischa Dohler. 2013. „Standardized Protocol Stack for the Internet of (Important) Things”. *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1389-1406.
- [5] Graeme M. Bragg, Kirk Martinez, Philip Basford, J. Hart. 2016. “868MHz 6LoWPAN with ContikiMAC for an Internet of Things environmental sensor network”. *2016 SAI Computing Conference (SAI)*, pp. 1273-1277.
- [6] Thomas Watteyne, Xavier Vilajosana, Branko Kerkez, Fabien Chraim, Kevin Weekly, Qin Wang, Steven Glaser, Kris Pister. 2012. “OpenWSN: a standards-based low-power wireless development environment”. *Transactions on Emerging Telecommunications Technologies*, 2012:23, pp. 480-493.