



Materiały, Technologie, Konstrukcje, Eksploatacja 2

Czaja J. (Redaktor)

Pre dykcja w ukł adach
mechanicznych
i automatycznych 2020 –
transmisja danych
pomiarowych

Wydano za zgodą Rektora

O p i n i o d a w c a

prof. dr hab. Janusz Szczepański
dr hab. Paweł Przybyłowicz, prof. AGH

R e d a k t o r n a c z e l n y

Wydawnictw Politechniki Rzeszowskiej
dr hab. inż. Lesław GNIEWEK, prof. PRZ

R e d a k t o r

Janusz Czaja

P r z y g o t o w a n i e m a t r y c

Patrycja Kuziora

P r o j e k t o k ł a d k i

Joanna Mikula

A u t o r z y

Adam Błachowicz

Leszek Klich

Adam Michałyszyn

sieci przemysłowe, sieci czujników, sieci bezprzewodowe, akwizycja danych

© Copyright by Oficyna Wydawnicza Politechniki Rzeszowskiej
Rzeszów 2020

Wszelkie prawa autorskie i wydawnicze zastrzeżone. Każda forma powielania oraz przenoszenia na inne nośniki bez pisemnej zgody Wydawcy jest traktowana jako naruszenie praw autorskich, z konsekwencjami przewidzianymi w *Ustawie o prawie autorskim i prawach pokrewnych* (Dz.U. z 2018 r., poz. 1191 t.j.). Autor i Wydawca dołożyli wszelkich starań, aby rzetelnie podać źródło zamieszczonych ilustracji oraz dotrzeć do właścicieli i dysponentów praw autorskich. Osoby, których nie udało się ustalić, są proszone o kontakt z Wydawnictwem.

ISBN 978-83-7934-443-7

Oficyna Wydawnicza Politechniki Rzeszowskiej
al. Powstańców Warszawy 12, 35-959 Rzeszów

Ark. wyd. 5,35. Ark. druk. 7,00.

Oddano do druku w grudniu 2020 r. Wydrukowano w grudniu 2020 r.

Drukarnia Oficyny Wydawniczej, al. Powstańców Warszawy 12, 35-959 Rzeszów

Zam. nr 99/20

Przedmowa redaktora

Morbum evitare quam curare facilius est (Lepiej zapobiegać niż leczyć) – fragment przysięgi Hipokratesa który z powodzeniem możemy odnieść do zagadnień opisywanych w niniejszej publikacji.

W pierwszej części zatytułowanej *Przewidywanie uszkodzeń maszyn z wykorzystaniem sieci przemysłowych i mechanizmów „Industry 4.0”* podjęto temat efektywnego zarządzania infrastrukturą zakładu przemysłowego. Jednym z elementów systemu zarządzania jest organizacja pracy Działu (komórki) Utrzymania Ruchu. W zakładach przemysłowych dysponujących rozległym (w dosłownym tego słowa znaczeniu) i zróżnicowanym parkiem maszynowym wyposażonym w urządzenia i linie technologiczne zawierające odpowiednie interfejsy We-Wy połączenie tych zasobów w sieć jest naturalnym działaniem menedżerów. Celem pierwszej części opracowania jest pokazanie studium przypadku firmy Pilkington Automotive Poland, jako swobodnego poradnika jak przygotować infrastrukturę aby móc w przyszłości elastycznie zarządzać konserwacjami i planowaniem prac utrzymania ruchu. W standardowym ujęciu zarządzanie zasobami realizowane jest poprzez działania prewencyjne (przeglądy, konserwacje) oraz reakcje doraźne (naprawy, remonty). Współczesne rozwiązania techniczne, oferowane przez systemy sterowania i kontroli zaimplementowane w nowoczesnych obrabiarkach i liniach technologicznych, pozwalają na wprowadzenie kolejnego narzędzia zarządzania zasobami jakim są mechanizmy przewidywania. Mechanizmy te pozwalają na redukcję nieoczekiwanych zatrzymań i bardziej niezawodne oraz elastyczne planowanie rutynowych zadań dla załogi utrzymania ruchu i pracowników produkcji. Jeżeli do tego dodamy możliwość podłączenia dużej ilości urządzeń do sieci, uzyskujemy efekt skali, który przy określonym progu ilości maszyn do obsługi, daje wymierne korzyści związane z oszczędnością czasu pracy służb utrzymania ruchu. W publikacji zaprezentowano topologię i elementy sieci, opisano sensory, elementy wykonawcze, urządzenia i oprogramowanie wizualizacyjne, bazodanowe i komu-

nikacyjne. Na podstawie danych zebranych z systemu monitorowania linii technologicznych podjęto próbę opracowania modelu matematycznego linii produkcyjnej. Takie modelowanie może być zastosowane do przewidywania stanu i kondycji maszyn. Predykcje obliczone na podstawie modelu mogą pomóc zautomatyzować decyzje i wygenerować wskazania dla operatorów i obsługi linii.

W kolejnym opracowaniu pt. *Budowa zdecentralizowanej sieci czujników bezprzewodowych na potrzeby akwizycji danych pomiarowych z linii produkcyjnych* przedstawiono jeden z podstawowych elementów wykorzystywanych w układach automatyki przemysłowej tj. budowę sieci opartej na czujnikach bezprzewodowych. Technologie bezprzewodowe pozwalają na łączenie w sieci o różnych topologiach coraz większej liczby różnego rodzaju sensorów pomiarowych, działających dzięki różnego rodzaju interfejsom o coraz większej dokładności pomiarów. Projektowanie systemów pomiarowych musi uwzględnić zapewnienie niezawodnych, redundantnych kanałów transmisji, które zapewnią wysoką niezawodność akwizycji danych. Kolejnym wyzwaniem stojącym przed projektantami systemów jest konieczność transmisji danych w sieciach rozległych i zapewnienie odpowiedniego poziomu elastyczności organizacji poszczególnych węzłów sieci. Stąd ukierunkowanie na technologie bezprzewodowe. W artykule opisano projekt bezprzewodowej akwizycji danych, wykorzystującej zabezpieczony tor transmisyjny przy wykorzystaniu szeroko dostępnych transceiverów RFM69HW połączonych w sieć. W projekcie zastosowano rozproszoną sieć czujników bezprzewodowych, pracujących w topologii kratowej i przesyłających przykładowe dane pomiarowe. W opracowaniu przedstawiono opis poszczególnych elementów sieci: transceiverów, czujników pomiarowych, anten i oprogramowania. Na zbudowanym prototypie przeprowadzono testy wybranych parametrów sieci oraz potwierdzono możliwość zbudowania sieci czujników, w której dodanie bądź uszkodzenie jednego lub n czujników spowoduje adaptację pozostałych węzłów do nowej liczby czujników i ich dalszą nieprzerwaną pracę na pozostałych węzłach.

W trzecim opracowaniu pt. *Budowa bezprzewodowej sieci czujników w oparciu o moduły Wi-Fi* przedstawiono charakterystykę modułu bezprzewodowego WiFi na przykładzie modułu ESP8266. Bardzo szczegółowo opisano budowę modułu, charakterystykę portów We-Wy, schemat blokowy modułu oraz podstawowe wersje modułu. Przedstawiono także tryby pracy modułu. W artykule opisano także charakterystykę różnych standardów

sieci WiFi oraz omówiono różne topologie sieci bezprzewodowych. W ramach eksperymentu zbudowano sieć opartą na jednej z wersji RFM69HW – module ESP-12. Do budowy przykładowej sieci użyto modułu rozwojowego NodeMCU V3. To platforma IoT typu open source, która zawiera oprogramowanie układowe działające na układzie ESP8266 Wi-Fi firmy Espressif Systems.

Podsumowując zagadania będące przedmiotem poszczególnych opracowań należy stwierdzić że przedstawione w nich rozwiązania techniczne zmierzają w kierunku jak najszerszego wykorzystania technologii Czwartej Rewolucji Przemysłowej określanej jako Industry 4.0 (Przemysł 4.0), złożonego procesu transformacji technologicznej i organizacyjnej przedsiębiorstw, który obejmuje m.in. cyfryzację produktów i usług. Wdrażanie tych rozwiązań możliwe jest dzięki wykorzystaniu nowych technologii cyfrowych, akwizycji danych, monitorowania zasobów oraz zapewnieniu w sieci współpracy maszyn, urządzeń i ludzi.

dr inż. Janusz Czaja
Politechnika Rzeszowska

Przedmowa Prezesa WSM



Każdy dąży do fortuny aby móc przewidzieć przyszłość i skutecznie na nią oddziaływać. Oczywiście pod warunkiem, że wydana fortuna zwróciłaby się z nawiązką. Ludzkość podejmowała działania w kierunku przewidywania przyszłości już od jej zarania. Wszystkim znane są np. przypadki wykorzystywania w starożytności wiedzy astronomicznej do sterowania decyzjami politycznymi. Jeszcze bliższym przykładem są prognozy pogodowe. Wprawdzie z dużą dokładnością można je przewidzieć i przygotować się do anomalii ale jeszcze na razie trudno na nie wpływać.

Metody predykcji są dzisiaj szeroko wykorzystywane w przemyśle. Jednym z obszarów gdzie ich znaczenie jest fundamentalne, są wielkoseryjne procesy produkcyjne. Każdy z zarządzających takimi procesami zawsze chce obserwować niezakłócony takt wytwarzania wyrobów na swojej linii produkcyjnej. Ale niestety nawet najlepiej zorganizowane procesy nie są stu procentowo skuteczne. Uważa się, że wskaźnik OEE (Overall Equipment Efficiency) na poziomie ponad 80% jest już efektywny. Jednym ze składników wpływających na rezultat OEE jest UT (Up Time), czyli dostępny czas pracy urządzeń produkcyjnych. Jedną z jego składowych jest z kolei TDT (Technical Down Time), czyli czas przestoju urządzeń z powodu awarii technicznych. Oczywiście zadaniem zarządzających procesami produkcji jest jego minimalizowanie. Można to robić w sposób reaktywny, czyli podejmować działania po wystąpieniu awarii, lub planowy, wyprzedzający pojawienie się awarii. Ten planowy sposób zawiera w sobie dwie metody: prewencyjną i predykcyjną. Zastosowanie każdej z tych trzech metod w określonych przypadkach ma swoje techniczne, technologiczne, a przede wszystkim ekonomiczne uzasadnienie. Tematyka tej monografii skupia się na wykorzystywaniu metod predykcyjnych w utrzymaniu ruchu

jako tych najbardziej skutecznych ale niekoniecznie w każdym przypadku najtańszych i mających powszechne zastosowanie. O finalnym ich użyciu zawsze decyduje analiza opłacalności.

Program predycyjnego utrzymania ruchu zainicjowany został przez Wschodni Sojusz Motoryzacyjny w ramach europejskiego programu DRIVES (Development and Research on Innovative Vocational and Educational Skills), którego celem jest przygotowanie programów kształcenia pod kątem przyszłych potrzeb branży motoryzacyjnej w Europie. Predycyjne utrzymanie ruchu ma charakter uniwersalny i ma zastosowanie we wszelkiego rodzaju firmach produkcyjnych. Wschodni Sojusz Motoryzacyjny i Wydział Mechaniczno-Technologiczny Politechniki Rzeszowskiej w Stalowej Woli podjęły się wprowadzenia tej tematyki do programu nauczania na poziomie studiów inżynierskich i magisterskich. Dzisiejsze wymagania dla skuteczności procesów produkcyjnych wymuszają coraz szersze stosowanie metod predycyjnych. Skoncentrowanie na Podkarpaciu takich branż produkcyjnych jak motoryzacja, lotnictwo czy przetwórstwo metalowe i spożywcze, gdzie występuje duże umaszynowanie, automatyzacja i robotyzacja, skutkuje zapotrzebowaniem na wysoko wykwalifikowanych specjalistów w tej dziedzinie. Do dalszego rozwoju metod predycyjnych przyczyni się na pewno wprowadzanie narzędzi przemysłu 4.0. Gromadzenie dużej ilości danych bezpośrednio z urządzeń i ich zautomatyzowana analiza to podstawa poprawnego wnioskowania i skutecznego podejmowania decyzji.

Jestem przekonany, że monografia *Predykcja w układach mechanicznych i automatycznych*, którą oddajemy do Waszej dyspozycji, przyczyni się do wzrostu poziomu wiedzy wykładowców, studentów i pracowników firm produkcyjnych, a w efekcie do podniesienia poziomu skuteczności procesów wytwórczych.

Ryszard Jania
Prezes Wschodniego Sojuszu Motoryzacyjnego

Badania oraz ich publikacja zostały zrealizowane dzięki środkom na utrzymanie potencjału badawczego Politechniki Rzeszowskiej w ramach umowy UPB.KI.20.001.

Badania zostały zrealizowane dzięki wykorzystaniu aparatury z projektu: *Utworzenie naukowo-badawczego Laboratorium Międzyuczelnianego w Stalowej Woli* w ramach Programu Operacyjnego Rozwój Polski Wschodniej 2007-2013, Oś Priorytetowa I – Nowoczesna Gospodarka, Działanie I.3 – Wspieranie Innowacji, numer projektu: POPW.01.03.00-18-016/12-00.

Spis treści

1	Predykcja uszkodzeń maszyn z wykorzystaniem sieci przemysłowych i mechanizmów „Industry 4.0”	
	<i>A. Błachowicz</i>	15
1.1	Wprowadzenie	16
1.2	Metody predykcyjne	20
1.3	Wyniki – przykłady modyfikacji oraz usprawnień wynikających z zastosowania mechanizmów predykcji	38
1.4	Dyskusja	41
1.5	Podsumowanie	42
	Bibliografia	43
2	Budowa zdecentralizowanej sieci czujników bezprzewodowych na potrzeby akwizycji danych pomiarowych z linii produkcyjnych	
	<i>L. Klich</i>	45
2.1	Wprowadzenie	46
2.2	Najważniejsze pojęcia	47
2.3	Założenia i model działania	52
2.4	Transceiver RFM69HW	57
2.5	Moduł pomiarowy (węzeł)	62
2.6	Pomiar napięcia akumulatora	66
2.7	Dobór anten	67
2.8	Oprogramowanie	69
2.9	Oprogramowanie pomiarowe i moduł pomiaru	71
2.10	Pomiary	73
2.11	Podsumowanie	75
	Bibliografia	77

3	Budowa bezprzewodowej sieci czujników w oparciu o moduły Wi-Fi	
	<i>A. Michajłyszyn</i>	79
3.1	Wprowadzenie	80
3.2	Charakterystyka modułu bezprzewodowego Wi-Fi na przykładzie ESP8266	82
3.3	Zasięg i jakość sygnału w zależności od zastosowanej anteny	87
3.4	Wi-Fi – charakterystyka i topologie	89
3.5	Charakterystyka bezprzewodowych sieci czujników i zastosowanie	92
3.6	Budowa przykładowej sieci Wi-Fi z zastosowaniem modułu ESP8266	94
3.7	Podsumowanie	103
	Bibliografia	103
	Spis rysunków	107
	Spis tablic	110

1. Predykcja uszkodzeń maszyn z wykorzystaniem sieci przemysłowych i mechanizmów „Industry 4.0”

ADAM BŁACHOWICZ¹

POLITECHNIKA RZESZOWSKA, A.BLACHOWICZ@PRZ.EDU.PL

Streszczenie Nowoczesne zakłady produkcyjne nieuchronnie zmierzają w kierunku standardów wyznaczanych przez pojęcie czwartej rewolucji przemysłowej. Wraz ze zmianą w maszynach, infrastrukturze i metodach zarządzania, zmienia się też podejście do pracy służb technicznych, w tym służb utrzymania ruchu oraz diagnostyki i predykcji uszkodzeń maszyn. W zakładach branży motoryzacyjnej, architektura sterowania maszyn i ich opomiarowanie pozwala na wdrożenie takich mechanizmów. Połączenie w sieć sterowników programowalnych zapewnia dostęp do danych, które można wykorzystać potem w algorytmach predykcji awarii. Ponadto nowoczesne sterowniki PLC, pozwalają na tworzenie zaawansowanych procedur (podprogramów) i implementację mechanizmów predykcji już na poziomie maszyny. Zgromadzone dane można wizualizować bezpośrednio na panelu maszyny, lub gromadzić w centralnej bazie, która połączona z portalem raportującym, stworzy gotowy system nadzoru i zarządzania parkiem maszynowym.

¹ORCID: 0000-0003-4465-2652, Wydział Mechaniczno-Technologiczny Politechniki Rzeszowskiej, Kwiatkowskiego 4, 37-450 Stalowa Wola

1.1. Wprowadzenie

Nowoczesne zakłady produkcyjne nieuchronnie zmierzają w kierunku standardów, wyznaczanych przez pojęcie czwartej rewolucji przemysłowej. Wraz ze zmianą w maszynach, infrastrukturze i metodach zarządzania, zmienia się też podejście do pracy służb technicznych. Jednym z ważniejszych działań pomocniczych, w każdym zakładzie produkcyjnym, jest dział Utrzymania Ruchu (UR). Jego efektywna praca opiera się na trzech filarach: reakcji, prewencji i predykcji. Pojawienie się w zakładach, urządzeń z interfejsami „IoT” (Internet of Things, dosł. „internet rzeczy”) oraz infrastruktury sieciowej na poziomie maszyn, pozwoliło wreszcie na uproszczenie dostępu do danych maszynowych i w efekcie na znaczący rozwój ostatniego z filarów pracy UR, a więc predykcji.

Główną przesłanką do stosowania predyktywnego utrzymania ruchu jest możliwość regularnego monitorowania aktualnego stanu maszyn i ich operacyjnej efektywności w procesie produkcji. Systemy predykcyjne powinny zapewnić maksymalne przedziały czasowe, pomiędzy naprawami maszyn i minimalizować liczbę oraz koszt ich nieplanowanego wyłączenia z produkcji [12].

W większości nowoczesnych firm produkcyjnych na świecie wdrożenie predykcyjnego utrzymania ruchu do systemu totalnego zarządzania fabryką optymalizuje dostępność maszyn w procesach i znacznie redukuje koszty utrzymania ruchu. W praktyce podejście predykcyjne staje się metodą prewencji sterowaną nie poprzez plany kalendarzowe, ale bezpośrednio przez mierzoną i kontrolowaną kondycję maszyn. Należy jednak pamiętać o tym, że predykcja nigdy nie będzie substytutem prewencji i nie zastąpi tradycyjnych metod konserwacji. Jest raczej wartościowym dodatkiem (wartością dodaną) do wszechstronnego programu zarządzania utrzymaniem ruchu w zakładach produkcyjnych. Tam, gdzie tradycyjne podejście polega na rutynowych przeglądach maszyn i szybkiej reakcji na awarie, mechanizmy predykcyjne pomagają planować specyficzne zadania dla pracowników utrzymania ruchu, które są wymagane przez aktualny stan maszyny i urządzenia w fabryce [10]. Pozwala to na redukcję nieoczekiwanych zatrzymań i bardziej elastyczne planowanie rutynowych zadań dla załogi utrzymania ruchu i pracowników produkcji.

W niniejszym artykule chciałbym, przedstawić techniki predykcji uszkodzeń maszyn, z wykorzystaniem mechanizmów sieciowych, w oparciu o stu-

dium przypadku działu Utrzymania Ruchu, fabryki szyb samochodowych NSG Pilkington Automotive Poland w Chmielowie. Przedstawione zostaną tutaj: struktura zastosowanych urządzeń automatyki i sterowania, ich liczba, nadrzędny system gromadzenia danych, oraz zalecana dla wykorzystania mechanizmów predykcji, architektura sieciowa.

1.1.1. Automatykacja w liczbach

Poziom automatykacji fabryki, można najlepiej przedstawić za pomocą liczb. W najnowszej fabryce NSG Pilkington w Chmielowie, która została wybudowana w 2012 roku, na powierzchni prawie 10 hektarów jest zlokalizowanych pięć wydziałów produkcyjnych. Najdłuższa linia ma długość 400 metrów.

Na całym obszarze pracuje:

- 40 robotów przemysłowych,
- 71 komputerów procesowych (systemy Scada i Supervisory),
- 121 sztuk sterowników PLC,
- 149 sztuk paneli HMI,
- 9 sztuk systemów wizyjnych.

Jak widać poziom automatykacji zakładów i linii produkcyjnych, zależy od zastosowanych urządzeń i mechanizmów sterowania. W nowoczesnych zakładach produkcyjnych urządzenia PLC zarządzają całą produkcją wykonując inteligentne i wielostrukturalne programy [9].

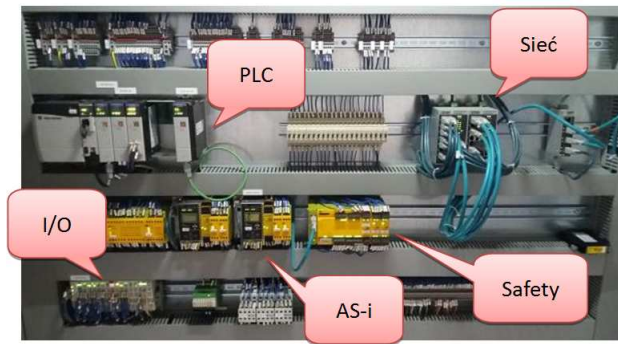
Software urządzeń z całej fabryki to około 250 GB, a backupy komputerów sterujących procesami wytwórczymi zajmują od 3 do 4 TB. Dla tych danych są na bieżąco wykonywane aktualne kopie zapasowe w ramach działań prewencyjnych.

Większość z tych urządzeń, jest połączona w sieć, co daje nam możliwość zdalnego odczytu wybranych danych pomiarowych i sygnałów z czujników. Dane te, są wyświetlane na ekranie, systemu monitoringu zakładu i logowane na potrzeby predykcji i analiz.

1.1.2. Architektura sterowania maszyn na liniach produkcyjnych

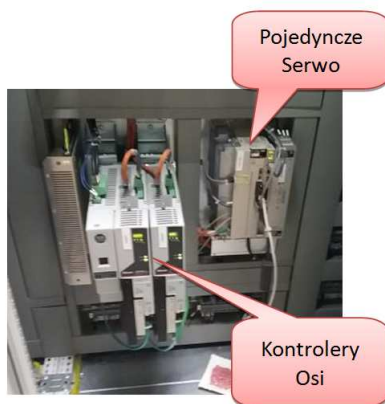
Maszyny i szafy sterujące, dla linii produkcyjnych wytwarzania szyb samochodowych, były tak projektowane, aby maksymalnie zestandaryzować użyte w nich urządzenia automatyki. Struktura szaf jest więc w dużej mierze podobna, co ma ogromne znaczenie dla służb UR w przypadku diagnostyki skomplikowanych awarii systemów sterowania.

„Sercem” lub raczej „mózgiem” każdej szafy sterującej jest kontroler PLC, wyposażony w dwie lub trzy karty sieciowe. Realizuje on logikę programu pracy całej linii i jej pojedynczych maszyn oraz odpowiada za komunikację. Obok niego mamy tam „switche” sieci Ethernet, „gatewaye” sieci AS-i, urządzenia bezpieczeństwa, pojedyncze serwonapędy i kontrolery osi dla manipulatorów wieloosiowych. Zazwyczaj drogie karty wejść i wyjść dyskretnych, zostały zastąpione rozproszonymi modułami I/O. Na Rysunkach 1.1 i 1.2 przedstawiono przykładowe rozmieszczenie elementów sterujących w szafie.



Rysunek 1.1: Elementy sterujące w szafie elektrycznej – część zarządzająca (logiczna).

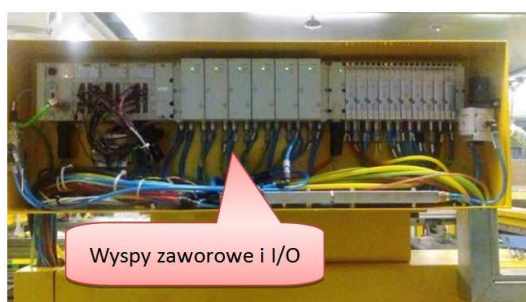
Jak widać sterowanie jest tu wyraźnie podzielone na dwie grupy: podzespoły logiki i sterowania oraz elementy wykonawcze. Znajduje to odzwierciedlenie w architekturze sieci przemysłowej Ethernet IP, która jest podzielona na dwa poziomy[13]: *górny*, nazywany *zarządzanym* („managed”), dla urządzeń logiki i sterowania (kontrolery PLC, komputery procesowe i panele HMI) oraz *dolny*, nazywany *niezarządzanym* („unmanaged”), dla urządzeń wykonawczych i bezpieczeństwa („gatewaye” i skanery sieci przemysłowych,



Rysunek 1.2: Elementy sterujące w szafie elektrycznej – część wykonawcza.

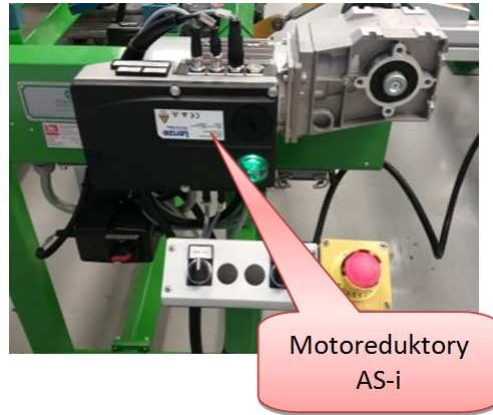
rozproszone moduły WE/WY, sieciowe moduły bezpieczeństwa, serwonapędy i urządzenia pomiarowe).

Osobnym zagadnieniem, są urządzenia automatyki zastosowane na maszynach „on board” (w ciągu produkcyjnym, poza szafą sterującą). Tutaj korzystamy z rozproszonych wysp i modułów ethernetowych I/O, a także napędów transportowych sterowanych z wykorzystaniem standardów sieci przemysłowych AS-i, Device-Net, Control-Net, Profibus lub Profinet (Rysunek 1.3).



Rysunek 1.3: Pneumatyczne wyspy zaworowe z interfejsem sieciowym (rozproszone).

Podsumowując, artykuł nasz możemy potraktować jako studium przypadku, które ukazuje w jaki sposób budować architekturę maszyn i urzą-



Rysunek 1.4: Falownik sieciowy silnika transportera.

dzeń oraz integrować je w sieć, aby potem móc zapewnić sobie dostęp do danych, które następnie można wykorzystać w predykcji awarii. Artykuł powstał w oparciu, o dokumentację techniczną, instrukcje urządzeń automatyki i sterowania firm Rockwell Automation, Festo i Biehl Wiedemann, oraz liczne opracowania naukowe. Wykorzystano tutaj informacje z instrukcji użytkowania (dokumentacji technicznej) oprogramowania wspomnianych wyżej firm i opracowań własnych działu UR fabryki Pilkington Automotive Poland w Chmielowie.

1.2. Metody predykcyjne

W tej części artykułu, chciałbym szczegółowo przedstawić zalecenia i wskazówki do budowy sieci przemysłowej (na przykładzie sieci Ethernet), opisać budowę serwera sieci przemysłowej, a w dalszej części omówić stosowane techniki predykcji. W szczególności te, które wykorzystują mechanizmy komunikacji sieciowej i algorytmy zagnieżdżone w rutynach programowych, sterowników PLC.

Podrozdziały 1.2.1 i 1.2.2 dotyczą architektury sieciowej i budowy serwera monitoringu danych predykcyjnych. Znajdziemy tutaj opis elementów, będących sprzętową częścią scentralizowanego systemu gromadzenia i przetwarzania danych. Kolejne rozdziały 1.2.3 do 1.2.6 opisują konkretne algorytmy i techniki predykcji, zastosowane w naszym systemie. Generalnie w przypadku wszechstronnych systemów predykcyjnych, powinny one

zawierać techniki diagnostyczne i dane dotyczące wibracji, termografii, parametrów procesu produkcyjnego, trybologii, testów ultradźwiękowych, inspekcji wizualnych i innych testów nieniszczących. Zostało to bardzo szczegółowo opisane w książce autorstwa R. Keith-Mobley „An Introduction to Predictive Maintenance” [10], która wyczerpująco opisuje klasyczne techniki predykcyjne. W naszym przypadku skupimy się na technikach wykorzystujących sieć przemysłową i sterowniki PLC. Dodatkowe i szczegółowo opisane informacje o takich algorytmach i technikach, można znaleźć w pozycjach [9], [17], [18] i [12].

Podrozdział 1.2.7 dotyczy odrębnego projektu predykcyjnego, polegającego na modelowaniu statystycznym fragmentu linii produkcyjnej i przewidywaniu wartości downtime’u technicznego. Model został zbudowany w oparciu o dane historyczne pozyskane z systemu monitoringu, na bazie których, przy użyciu regresji, otrzymano zależność pozwalającą na predykcję poziomu czasu przestojów [2].

1.2.1. Sieci przemysłowe i ich zalecana struktura (topologia)

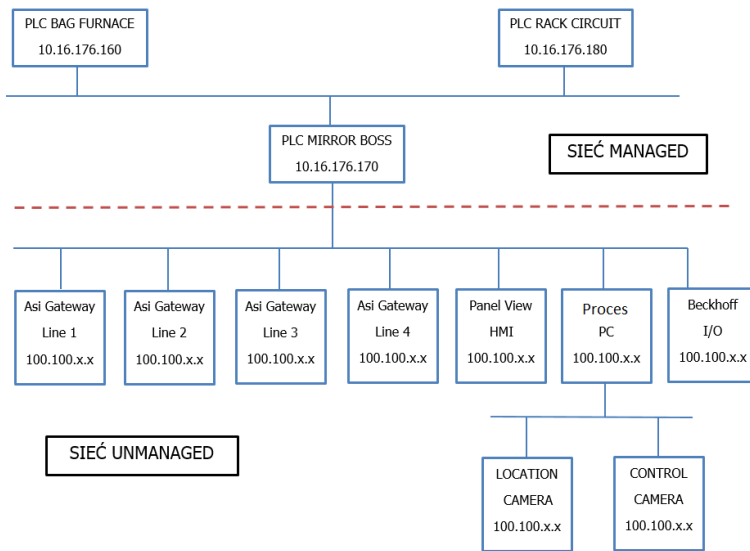
Podstawowym i koniecznym warunkiem rozpoczęcia budowy nowego lub transformacji już istniejącego zakładu do poziomu odpowiadającego wymaganiom czwartej rewolucji przemysłowej jest jego integracja sieciowa. Urządzenia sterujące muszą być połączone ze sobą na odpowiednich poziomach, zapewniających niezawodne przesyłanie i wymianę sygnałów oraz informacji.

W naszym zakładzie podstawową siecią jest Ethernet. Jego topologia jest wielopoziomowa i rozdzielona na kilka podgrup adresowych [13]. Mamy tutaj do czynienia z siecią przemysłową zarządzalną, połączoną ringiem światłowodowym, do której są podłączone sterowniki PLC. Ta sieć, poprzez firewall łączy się z siecią korporacyjną, przez którą wybrane i autoryzowane osoby mogą sprawdzać informacje o statusie pracy, wybranych linii produkcyjnych.

Sterowniki PLC są wyposażone w kilka kart sieciowych (zazwyczaj dwie). Jedna to karta, przez którą sterownik łączy się z siecią zarządzalną (wysokiego poziomu), natomiast druga z nich tworzy interfejs dla sieci podrzędnej każdego sterownika, w której znajdziemy: gatewaye AS-i, rozproszone wyspy I/O, serwonapędy, inwertery, panele, roboty i komputery procesowe (wyjątkiem są systemy SCADA podłączone do sieci wysokiego poziomu).



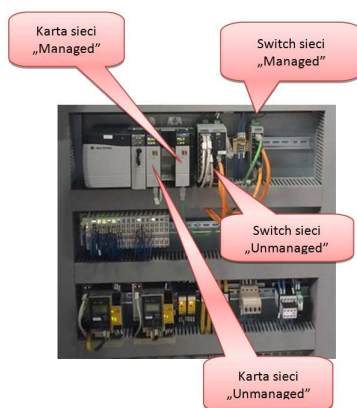
Rysunek 1.5: Switchce zarządalne („managed”) sieci wysokiego poziomu.



Rysunek 1.6: Topologia sieci Ethernet z rozdziałem na dwa poziomy (logiczny i wykonawczy).

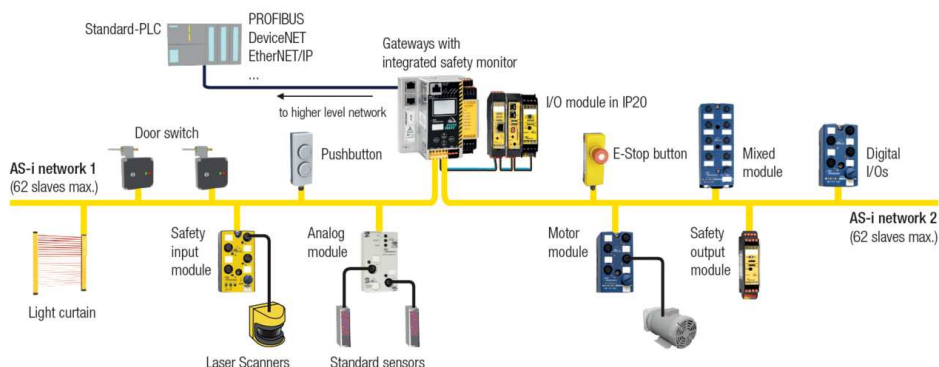
Z punktu widzenia topologii sieciowej, sterownik PLC jest punktem granicznym, pomiędzy sieciami: *zarządzalną* (wysokiego poziomu, zwaną u nas w wewnętrznym slangu „Managed”) zawierającą elementy sterujące realizujące logikę pracy linii oraz siecią *podrzedną* („Unmanaged”) zawierającą elementy wykonawcze. Pozwala na to jego modułowa budowa, dzięki której, w kasecie sterownika, możemy umieścić wiele kart sieciowych. Jest to rozwiązanie stosowane często w wielu zakładach, które wykorzystują tzw.

Master PLC do gromadzenia danych maszynowych na potrzeby systemów MES. W naszym przypadku, nie korzystamy z Master PLC, natomiast zbudowaliśmy komputerowy system monitorujący pracę zakładu i gromadzący dane maszynowe tzw. „Live Data Server” (patrz podrozdział 1.2.2).



Rysunek 1.7: Punkt podziału sieci wysokiego i niskiego poziomu.

Uzupełnieniem sieci Ethernet jest sieć AS-i. Łączy ona napędy transporterów, rozproszone WE/WY i urządzenia bezpieczeństwa bezpośrednio na maszynach. Poniżej (Rysunek 1.8) przedstawiono schemat sieci AS-i zaopieczony z materiałów producenta podzespołów, firmy Bihl-Wiedemann [1]. Taki układ jest powszechnie stosowany.



Rysunek 1.8: Schemat sieci przemysłowej AS-i.

Oprócz wspomnianych wyżej sieci, używamy również standardów: Device-net, Profinet, MPI/Profibus, Control-net.

Zastosowanie sieci komunikacyjnych, ich wielopoziomowa struktura i topologia, pozwalają nam na dostęp do informacji i danych, na różnych poziomach maszyny czy całej linii produkcyjnej. Jest to istotne z punktu widzenia bieżącego monitoringu pracy i parametrów maszyn.

Bazując na opracowaniu [7] możemy stwierdzić, że wykorzystanie predykcji w układach sieciowych pomaga w szybszej detekcji nieprawidłowości i pozwala na lepsze wykorzystanie danych. Korzyści z takiego podejścia wpływają na poprawę efektywności systemów, jakości serwisu, zwiększenie zdolności do wykrywania nieprawidłowości i adaptację sprawdzonych algorytmów do nowych urządzeń. W przyszłości pozwoli to na połączenie technik predykcji z uczeniem maszynowym i optymalizacją procesów w sieci urządzeń, których awaryjność jest procesem losowym (stochastycznym).

1.2.2. Monitoring i gromadzenie danych w dziale UR – serwer sieci przemysłowej

Jednym z narzędzi, bieżącego nadzoru nad stanem fabryki (obok systemu SAP) jest serwer monitoringu parametrów maszyn i linii produkcyjnych na warsztacie Utrzymania Ruchu (Rysunek 1.9). Jest to system podłączony do sieci zarządczej i zbierający informacje ze sterowników PLC linii oraz urządzeń, sterujących zasilaniem zakładu w media. Jest to tzw. Live Data Server, czyli komputer z systemem operacyjnym Windows Server i zainstalowanym specjalistycznym oprogramowaniem komunikacyjnym i wizualizacyjnym. System pozwala na generowanie wykresów i trendów dla wielu danych, kojarząc je na jednym wykresie i umożliwiając wnioskowanie o pracy maszyn i urządzeń, np. dla różnych warunków atmosferycznych czy produkowanych modeli i parametrów. Obecnie korzysta on z komercyjnego oprogramowania firmy Rockwell Automation, zintegrowanego z siecią zakładową i dopasowanego do architektury linii produkcyjnych fabryki. Składniki systemu to:

- oprogramowanie wizualizacyjne Factory Talk View SE [14],
- baza danych maszynowych Factory Talk Historian [15],
- portal raportujący Factory Talk Vantage Point [16],
- oprogramowanie komunikacyjne KEP Server Enterprise,

- oprogramowanie komunikacyjne RSLinx: Enterprise i Classic Gateway.



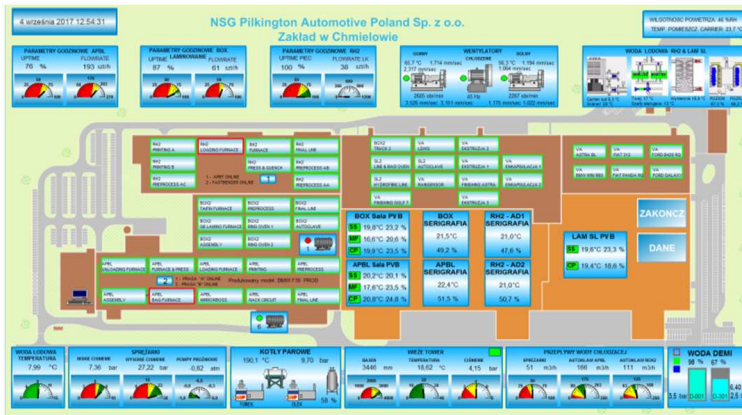
Rysunek 1.9: System monitoringu pracy zakładu.

Ekran główny systemu zawiera najważniejsze informacje o statusie pracy poszczególnych linii i o stanie urządzeń strategicznych w zakładzie, wytwarzających na potrzeby produkcji takie media jak:

- sprężone powietrze,
- para technologiczna,
- woda chłodząca autoklawy,
- woda lodowa do klimatyzacji,
- woda zdemineralizowana.

System ten oprócz bieżącego monitoringu, umożliwia bardzo szybką detekcję obszarów do wdrożenia działań prewencyjnych i wyprzedzających, oraz logowanie danych historycznych na potrzeby prowadzenia analiz RCA. (Root Cause Analysis). Drugim ważnym zadaniem systemu jest wizualizacja danych na wykresach, celem predykcji awarii dla wybranych przez Inżynierów Linii, kluczowych maszyn i węzłów (Rysunek 1.10).

Selekcja krytycznych parametrów maszyn i urządzeń, wymaganych do monitoringu, jest mocno skomplikowana z punktu widzenia stacjonarnego

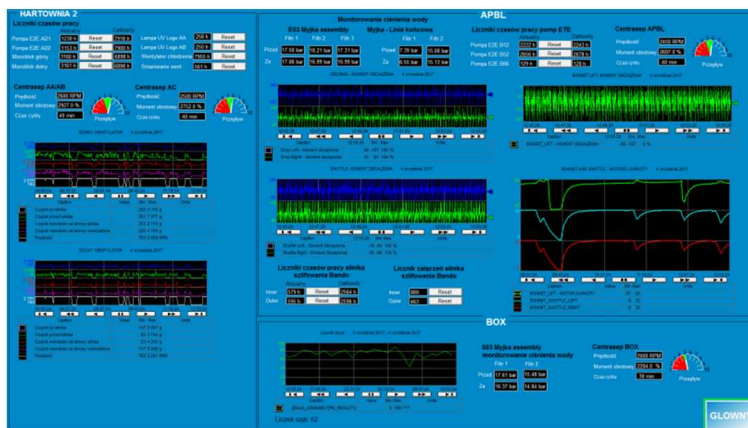


Rysunek 1.10: Ekran główny systemu monitoringu.

i scentralizowanego systemu sieciowego. Dostępne techniki i urządzenia predykcyjne działają w większości jako rozwiązania lokalne. Dlatego wybierając do analizy konkretną technikę, urządzenie lub obszar, musimy być przekonani, że będziemy monitorować urządzenia, o kluczowym znaczeniu dla produkcji [10]. Założeniem wdrażania scentralizowanego systemu predykcyjnego jest kombinacja różnych technik lokalnych i łączenie ich we wspólnej sieci. Każda fabryka czy zakład produkcyjny, musi zdecydować, jaka kombinacja technik predykcyjnych będzie wymagana dla jego urządzeń i zapewni optymalne efekty w aspekcie kosztów i niezawodności.

W opisywanym przypadku fabryki szyb samochodowych, do systemu monitoringu wybrane zostały sygnały i dane, związane z tzw. procesami „secondary” obróbki szkła. Są to procesy gięcia i laminowania, zakwalifikowane jako kluczowe i strategiczne dla produkcji szyby samochodowej (Rysunek 1.11).

Serwer ten daje pracownikom utrzymania ruchu, już na poziomie brygady reakcyjnej, możliwość globalnego spojrzenia na całą fabrykę. Dzięki temu poprawiają się możliwości diagnostyczne, a dodatkowo bieżąca obserwacja trendów dla kluczowych parametrów stała się codzienną rutyną operatorów. Pozwala to na poprawę niezawodności maszyn i przyspiesza reakcję na odchylenia od normalnego stanu monitorowanych zmiennych. Innymi słowy obserwacja trendów dla pomiarów i ich krytycznych wartości stała się podstawową techniką predykcji możliwą dzięki zastosowaniu sieci komunikacyjnej i doprowadzeniu sygnałów bezpośrednio do systemu,



Rysunek 1.11: Ekran logowanych danych predykcyjnych.

zlokalizowanego na warsztacie utrzymania ruchu.

1.2.3. Algorytmy predykcyjne zagnieżdżone w sterownikach PLC

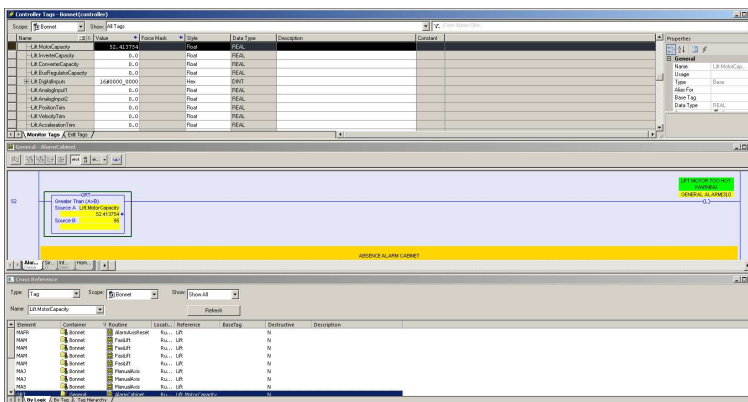
Nowoczesne sterowniki PLC, wyposażone w mechanizm niezależnych bloków lub rutyn programowych, pozwalają na tworzenie podprogramów, których zadaniem jest monitoring wybranych parametrów (zmiennych) oraz przetwarzanie ich i wystawienie sygnałów ostrzegawczych lub alarmowych w przypadku przekroczenia ustalonych granic już na poziomie monitorowanej maszyny, obrabiarki lub urządzenia technologicznego.

Mierzone wartości, np. drgań czy momentu obciążenia, podłączone są na wejście sterownika. Dedykowany podprogram czytuje je, poddaje skalowaniu i potem obliczeniom aktualnych wartości. Tak przetworzony wynik porównuje się do zdefiniowanych granic, ostrzegawczej lub alarmowej. W przypadku osiągnięcia jednej z nich na wyświetlaczu panelu operatorского pojawia się stosowny komunikat.

Najważniejszym elementem skutecznej predykcji jest takie dobranie wartości granicy ostrzegawczej, tzw. „warningu” aby w bezpiecznym horyzoncie czasowym, minimum 8 godzin, wyprzedzić wystąpienie komunikatu alarmowego, czyli już de facto uszkodzenia. Granice te dobiera się na podstawie analizy zgromadzonych danych historycznych, które są niezbędną bazą do predykcji. Sam sterownik PLC nie posiada zasobów pamięci do gromadze-

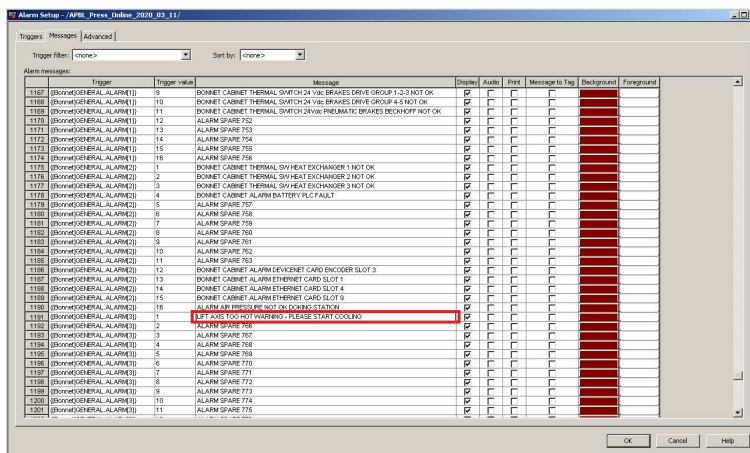
nia dużych ilości pomiarów, stąd niezbędne jest zbieranie tych danych na zewnętrznym systemie bazodanowym, np. lokalnym serwerze, znajdującym się w tej samej sieci co sterownik PLC. Dopiero tam obserwując trendy pomiarów i ich korelację z rzeczywistym wystąpieniem awarii można dobrać właściwą wartość graniczną, którą należy umieścić w algorytmie PLC, aby nasz mechanizm predykcyjny był skuteczny. Działa tu mechanizm podobny do sprzężenia zwrotnego, gdzie do zdefiniowanego surowego algorytmu wprowadzamy wartość oszacowaną w zewnętrznym systemie.

Przykładem takiego algorytmu jest predykcja uszkodzenia mechanicznego układu napędu opuszczania formy do przenoszenia ukształtowanych szyb przednich, tzw. „Bonnet.liftu”. Sterownik PLC monitoruje na bieżąco dane z momentów obciążenia silników serwo napędu układu i ich temperatury. Dane są skalowane do znormalizowanych jednostek i porównywane do granicy ostrzegawczej, która w naszym przypadku wynosi 95%, tzw. „Motor capacity”. Cały proces odbywa się w sposób ciągły, a program odświeża wartości pomiarowe co 14ms. Istotnym aspektem jest tutaj filtrowanie momentu obciążenia podczas startu układu, gdyż rozruch silników wiąże się z nagłym wzrostem prądu i momentu. Najważniejszym parametrem jest pomiar amplitudy momentu obciążenia układu już po osiągnięciu prędkości przejazdu (po zakończeniu akceleracji).



Rysunek 1.12: Logika obsługi ostrzeżenia, przekroczenia parametru granicznego silnika serwo w programie drabinkowym RSLogix 5000, sterownika PLC.

Więcej szczegółowych informacji dotyczących metod predykcji uszkodzeń w układach napędowych z wykorzystaniem sterowników PLC można



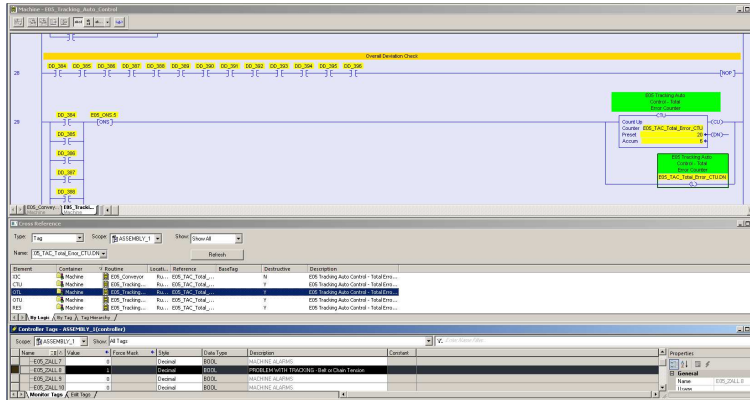
Rysunek 1.13: Komunikat ostrzeżenia wyświetlany na ekranie pulpitu operator-
skiego. Tutaj definicja komunikatu w programie Factory Talk View Studio.

znaleźć w pozycji [17]. Przy czym postawiono tam tezę, że sterownik nie jest najlepszym urządzeniem do zastosowania jako kontroler predykcyjny, ze względu na swoją niewielką moc obliczeniową. Dlatego dane najlepiej transferować do zewnętrznego systemu, np. komputera, który zrealizuje wszystkie obliczenia na potrzeby predykcji.

Drugim przykładem algorytmu predykcyjnego, zagnieżdżonego w pod-
programie sterownika PLC jest detekcja uszkodzenia mechanicznego układu
napędu myjki tunelowej do szyb przed laminowaniem. Ten algorytm wyko-
rzystuje mechanizm śledzenia obiektu (szyby), w porównaniu z sygnałem
impulsowym ruchu łańcucha napędowego przenośnika szyb.

Obiekt musi pojawić się w określonym miejscu, w określonym i zdefinio-
wanym „oknie” wartości impulsów zainstalowanego kodera (impulsatora).
Wraz z postępującym zużyciem układu napędowego, tj. wyciąganiem łań-
cucha, luzem na kołach pasowych i uszkodzeniem łożysk zmienia się war-
tość szerokości okna pojawienia się obiektu. Na podstawie analizy histo-
rycznych danych, wyznaczone zostały takie wartości szerokości impulsowej
okna pomiarowego, które pozwalają na wcześniejsze wyświetlenie komuni-
katu o zbliżającym się uszkodzeniu układu napędowego i konieczności jego
przeglądu.

Nadmienić tu należy, że układ napędowy pracuje w środowisku wody



Rysunek 1.14: Logika ostrzeżenia, przekroczenia parametru granicznego okna impulsów w programie drabinkowym RSLogix 5000, sterownika PLC. Blok licznika przekroczeń impulsów okna.

zdemineralizowanej, gdzie zużycie korozyjne elementów metalowych jest nieregularne. Utrudnia to pracę w oparciu tylko o prewencję, gdyż w przeszłości, zdarzało się często wymieniać elementy za późno. Informacja pojawiająca się na panelu operatora nie jest de facto alarmem i nie zatrzymuje maszyny. Jej zadaniem jest wywołanie określonej reakcji przez obsługę (w tym wypadku wezwanie służb utrzymania ruchu). Wprowadzony algorytm predykcyjny pozwala na bardziej elastyczne i świadome zarządzanie konserwacją urządzenia.

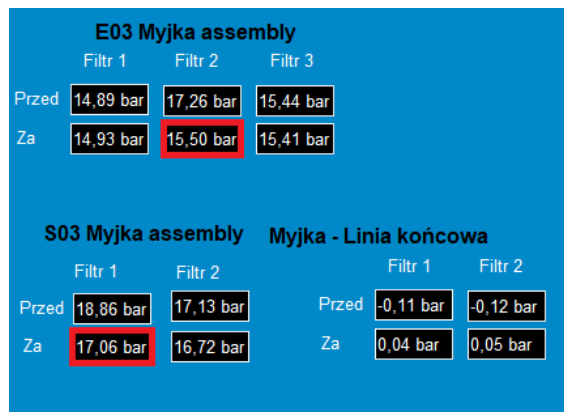
Przedstawione powyżej metody wykorzystują standardowe kalkulacje oraz procedury programowe w sterownikach PLC, działające w większości w oparciu o porównania i odniesienia do wzorców. Przyszłością rozwoju predykcji, zagnieżdżonej w sterownikach jest wykorzystanie mechanizmów sieci neuronowych, które są predykcyjne z natury [9]. Zaawansowane sterowniki z rozszerzoną pamięcią programu pozwalają na wprowadzanie kodu w językach listy instrukcji, a to pozwala na zagnieżdżanie w nich algorytmów opartych o mechanizmy sieci neuronowej. Mogą one rygorystycznie monitorować proces, sygnały z czujników a także sam przebieg wykonania poszczególnych etapów programu [9].

Opisane powyżej algorytmy zostały wdrożone na sterownikach serii Controllogix 5000 firmy Rockwell Automation. Oczywiście można stosować je w każdym innym sterowniku PLC, a jedynym warunkiem zastosowania jest wystarczająca ilość pamięci programu i zastosowanie języka programowania

na systemie monitoringu dla pracowników utrzymania ruchu, na warsztacie odległym od maszyny o kilkaset metrów.

1.2.6. Pomiary ciśnień urządzeń filtrujących

Bardzo duże ilości filtrów powietrza, stosowanych na myjkach i urządzeniach klimatyzacji pomieszczeń czystych, powodują wysokie koszty wymiany i konieczność absorbowania dużej ilości pracowników do przeprowadzenia tych prac. W miejsce cyklicznej prewencyjnej wymiany, zastosowano predycyjne podejście do testowania filtrów poprzez pomiar ciśnienia powietrza przed i za filtrem. Pozwala to na oszacowanie stanu zanieczyszczenia filtra i podjęciu decyzji, o jego wcześniejszej lub późniejszej wymianie.



Rysunek 1.19: Monitoring wartości ciśnień.

W sytuacji wstępnego oczyszczania powietrza, na hali produkcyjnej, zastosowanie predycyjnego pomiaru ciśnień pozwoliło na wydłużenie cyklu pracy filtrów na jednostkach klimatyzacji pomieszczeń malowania i laminowania szyb, co przyniosło wymierne oszczędności. I tutaj również z pomocą przyszła nam integracja pomiarów z siecią przemysłową i możliwość „sprawdzenia” ich do wizualizacji na warsztacie utrzymania ruchu. Operator UR ma w każdej chwili wgląd w pomiary parametrów pracy krytycznych dla produkcji filtrów i gdy tylko system wykryje przekroczenie zdefiniowanej wartości różnicy ciśnień, podejmuje decyzję o ich wymianie.

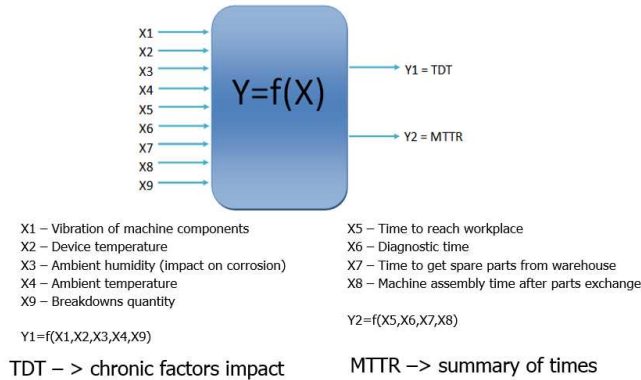
1.2.7. Modelowanie statystyczne awarii maszyn

Predykcja pojedynczych elementów czy urządzeń, bazuje na pomiarach określonych wielkości fizycznych. Mamy tutaj wyraźnie zdefiniowane granice kontrolne, warunki destrukcyjne, określone przez producenta warunki pracy i wynikające z nich parametry niezawodności jak np. MTBF. Co w sytuacji, gdy musimy określić te parametry dla większej grupy urządzeń, połączonych w linię produkcyjną? Teoretycznie dla szeregu urządzeń niezawodność całości wyznacza niezawodność najsłabszego elementu. Wynikałoby z tego, że niezawodność całości jest iloczynem niezawodności składników. Niestety ze względu na zmienność warunków pracy, wynikającą chociażby z różnorodności produkowanych asortymentów i różnych recept na linii – poszczególne elementy nie są zawsze jednakowo obciążone. Z pomocą w takiej sytuacji przychodzi nam statystyczne modelowanie, na podstawie którego, możemy zbudować matematyczny model linii produkcyjnej.

Mając taki model i określone dla niego zależności wejść i wyjść, możemy przewidywać poziom szukanego wskaźnika w zależności od zadanych wartości wejściowych. Dla utrzymania ruchu maszyn i urządzeń, kluczowym wskaźnikiem jest „Downtime” techniczny. Gdy będziemy mogli go zamodelować, dla wybranej linii produkcyjnej, możemy na tej podstawie przewidywać jego wartość dla różnych sytuacji, spowodowanych zmianami czynników wejściowych. Tym samym przenosimy predykcję stanu naszych urządzeń na wyższy poziom, zdefiniowany przez model naszej linii produkcyjnej.

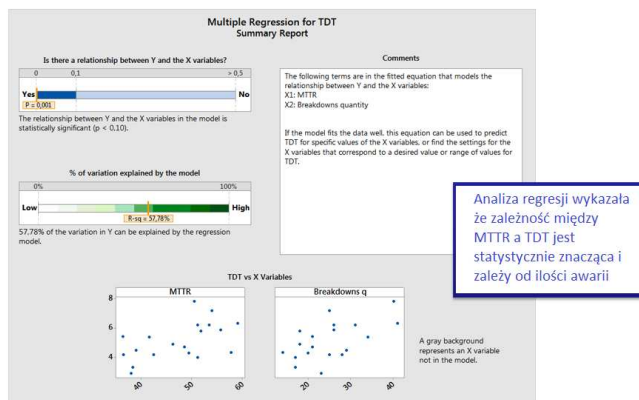
Takie modelowanie może być zastosowane do przewidywania stanu i kondycji maszyn, ponadto stanowi gotowy materiał do wdrażania zaawansowanych mechanizmów (np. uczenia maszynowego), które mogą automatyzować decyzje i generować rekomendacje dla osób zarządzających [4].

Poniżej został przedstawiony przykładowy model odcinka linii produkcyjnej (maszyny), utworzony na podstawie danych historycznych, dla którego została zdefiniowana zależność matematyczna pozwalająca na predykcję poziomu technicznego downtime’u (Rysunek 1.20). Do analizy danych i budowy modelu zastosowano oprogramowanie Minitab w wersji 17. Modelowanie było przeprowadzone w ramach projektu 6Sigma wdrażanego na obszarze utrzymania ruchu [2].



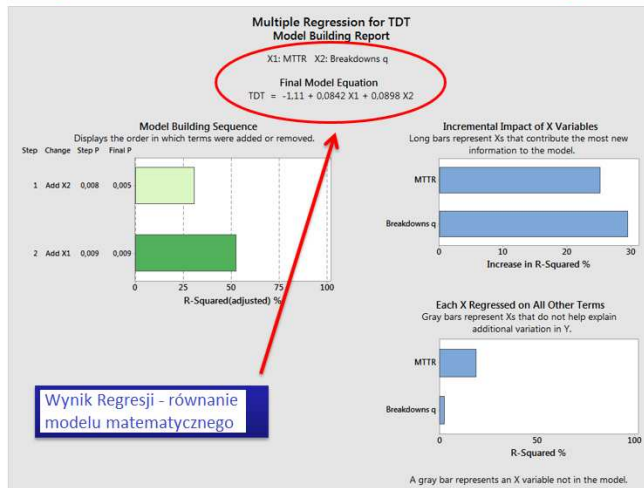
Rysunek 1.20: Wstępna definicja modelu maszyny do predykcji poziomu technicznego downtime'u.

Na początku zdefiniowano prosty wstępny model, dla którego zostały określone zmienne wejściowe i wynikowe. Następnie wszystkie mierzalne historyczne dane, zostały poddane testowi na zgodność z rozkładem normalnym i dopiero to było warunkiem dopuszczającym je do modelowania [3]. Zgromadzone dane zostały poddane analizie regresji, na podstawie której określono zależności pomiędzy poszczególnymi zmiennymi i wyłoniono te szczególnie znaczące (Rysunek 1.21).



Rysunek 1.21: Wynik regresji dla jednego z wyjść modelu.

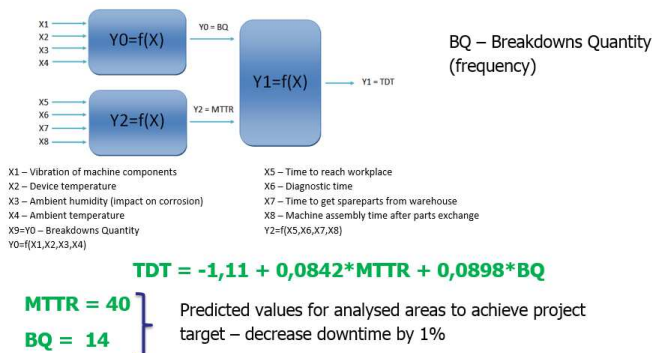
Kolejnym krokiem było określenie matematycznej zależności, pomiędzy wyłoniłymi statystycznie znaczącymi zmiennymi (Rysunek 1.22).



Rysunek 1.22: Wynik modelowania jako równanie matematyczne.

Wynikiem tego działania jest równanie matematyczne, pozwalające na wyliczanie poziomu technicznego downtime'u maszyny dla poszczególnych poziomów wartości wejściowych X. W trakcie analiz danych naszego projektu i kalkulacji regresji, zamodelowana pierwotnie zależność została zweryfikowana i na podstawie wyników analizy zmodyfikowano definicję modelu. Prawidłowy przedstawia się następująco (Rysunek 1.23).

According to regression results process model should be redefined:



Rysunek 1.23: Zredefiniowany model maszyny.

Dobrze zdefiniowany i „wytrenowany” model funkcji musi być zdolny do wyliczenia możliwie jak najdokładniejszego wyniku predykcji, dając nam nieznanne wcześniej, przyszłe wartości Y , wyliczone dla instancji wejściowych X [4]. Zależnie od tego możemy sami wybrać właściwy, specyficzny algorytm, zależny od rozmiaru i formatu dostępnego zestawu danych. Pośród dostępnych algorytmów, obok zastosowanej w naszym przypadku regresji, mamy do dyspozycji także: drzewo decyzji, lasy losowe i zaawansowane algorytmy uczenia maszynowego jak, np. wzmocnienie gradientowe [4].

Modelowanie statystyczne pozwala nam, na predykcję poziomu downtime’u technicznego urządzenia bądź odcinka linii produkcyjnej, w zależności od wartości wejść, które są opomiarowane i logowane w systemie sieciowym. Analizując taki model w drugą stronę, możemy (znając wartości technicznego downtime’u jako zagregowanego wyniku) kontrolować stan opomiarowanych wejść, a więc rzeczywistej kondycji maszyn, będących częściami składowymi całej linii. Podejście to jest zbieżne z metodologią zarządzania produkcją, opartą o monitoring i ciągłą kontrolę kluczowych wskaźników.

Wyniki modelowania predykcyjnego maszyn, pokazują duży potencjał do identyfikacji typów uszkodzeń i sposobów (trybów) ich powstawania. Niektóre zagadnienia pozwalają na zastosowanie algorytmów statystycznych, jak np. regresji lub korelacji, jednak zaawansowane modele wymagają wykorzystania rozwiązań bazujących na sztucznej inteligencji [8]. Systematyczne podejście z użyciem Big Data czy uczenia maszynowego pozwoli w przyszłości na poprawę szybkości i dokładności działania algorytmów predykcyjnych, co przełoży się bezpośrednio na strategię podejmowania decyzji w działach Utrzymania Ruchu.

1.3. Wyniki – przykłady modyfikacji oraz usprawnień wynikających z zastosowania mechanizmów predykcji

1.3.1. Usprawnienia maszyn

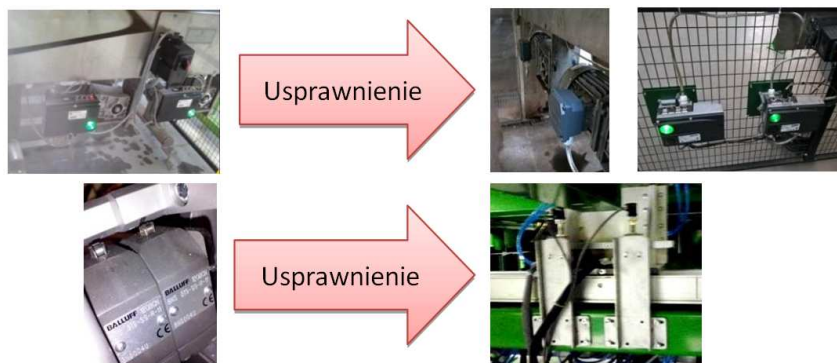
Obserwacja danych serwera monitoringu i analiza przyczyn awarii, doprowadziły zespół specjalistów utrzymania ruchu, do wniosków, że znaczny procent awarii, występuje w miejscach, gdzie maszyny są długotrwale narażone na chroniczne czynniki destrukcyjne. Takimi czynnikami mogą być:

podwyższona temperatura, wilgoć, brud, wibracje czy zużycie mechaniczne wynikające z tarcia. I nie ma w tym nic odkrywczego, gdyż są to stwierdzenia zgodne z teoriami TPM czy też Lean Maintenance.

Monitoring stanu tych czynników stanowi więc podwaliny pod predykcyjne podejście do zarządzania parkiem maszynowym i wdrożenia tzw. „*Condition Based Maintenance*”. Pracując zgodnie z takim podejściem dużo łatwiej dochodzimy do przyczyn niesprawności maszyn i możemy je wyeliminować u źródła.

Poniżej chciałbym pokazać dwa z naszych przykładowych rozwiązań takich problemów (niech służą jako tzw. „*Lesson Learnt*”):

- 1) Motoreduktory z inwerterami, pracujące w środowisku podwyższonej wilgotności – usunięcie inwerterów z silników i przeniesienie ich na siatkę w miejscu suchym.
- 2) Wymiana czujników najazdowych z kontaktem mechanicznym na czujniki indukcyjne bezkontaktowe.



Rysunek 1.24: Przykłady usprawnień na maszynach.

W omawianym przez nas przypadku, ważne jest to, że scentralizowany system nadzoru pracy maszyn, potwierdził skuteczność metodologii TPM, CBM czy Lean Maintenance. Ponadto pozwolił na odkrycie tego ludziom, którzy o tych metodologiach nigdy wcześniej nie słyszeli i spotkali się z nimi pierwszy raz w nowoczesnym zakładzie motoryzacyjnym (spełnił więc też funkcję edukacyjno-szkoleniową). Potwierdza to tylko, jak duży potencjał tkwi w rozbudowanych sieciowych systemach monitoringu parametrów ma-

szyn i że wdrożenie takich systemów to niezbędny krok do upowszechnienia predykcijnego podejścia w pracy działów utrzymania ruchu.

1.3.2. Rozwój działu Utrzymania Ruchu

Praca zespołu Utrzymania Ruchu w omawianym przykładzie firmy NSG Pilkington Automotive Poland, to nie tylko naprawy, predykcja, konserwacje i akcje prewencyjne. Pracownicy biorą również udział w projektach rozwoju automatyzacji w zakładzie. Sytuacja na rynku pracy i wymagania konkurencyjności w przemyśle motoryzacyjnym, wymuszają konieczność udziału w projektach zwiększania wydajności urządzeń (poprzez optymalizację ich programów) i automatyzacji stanowisk pracy fizycznej. W pierwszym przypadku mówimy o skracaniu czasu cyklu poszczególnych maszyn i eliminowaniu niepotrzebnych przestojów, w drugim o opracowaniu koncepcji maszyn i systemów, mających zastąpić ludzi, przy niektórych operacjach technologicznych [11].

W samym dziale UR działania rozwojowe i optymalizacyjne skupiają się na rozszerzaniu wdrożonych mechanizmów predykcyjnych na nowe obszary i maszyny. Drugą część działań to przygotowanie serwera danych maszynowych do roli portalu raportującego parametry produkcyjne. Chodzi o to, aby obok monitoringu parametrów technicznych maszyn, monitorować też proces wytwórczy. W dalekim horyzoncie czasowym te działania miałyby na celu przygotowanie serwera i sieci zakładowej do współpracy z systemami MES i ERP.

W tym celu, na istniejącym systemie, została wdrożona instalacja oprogramowania bazodanowego i portalu raportującego, opartego o mechanizm webowy, jako narzędzia do prezentacji i analizy danych procesowych. Na ich podstawie, możliwe będzie wyciąganie wniosków i podejmowanie decyzji, przez managerów, zarówno w aspekcie technicznym jak i produkcyjnym.

Kwestia predykcji zostanie wtedy rozszerzona z obszaru utrzymania ruchu, na obszar produkcji, gdzie będzie możliwe zastosowanie jej algorytmów do przewidywania powstawania wad w wytwarzanym produkcie.

Należy jednak pamiętać, że implementacja systemu predykcijnego jest kosztowna. Do zakupu instrumentów pomiarowych dochodzą koszty związane z ich instalacją i integracją z lokalnymi sterownikami. Dodatkowo system wymaga zarządzania i absorbuje pracę specjalisty, co też stanowi ob-

ciężenie finansowe.

Aby z sukcesem wdrożyć program predykcyjny musimy określić zyski, które będzie on w stanie wygenerować [10]. Ten cel da się osiągnąć, jeżeli:

- założenia programu opierają się, o kalkulacje mierzalnych zysków,
- zastosowano odpowiednie techniki predykcyjne,
- wysiłek poniesiony na wdrożenie, nie przekroczy wysiłku ponoszonego na standardową prewencję.

1.4. Dyskusja

Analizując przedstawione mechanizmy i osiągnięte rezultaty należy podkreślić aspekt wykorzystania infrastruktury sieci komunikacyjnych do wdrożenia mechanizmów predykcyjnych. W tym wypadku nie chodziło o skomplikowane metody i szczegółowe analizy, możliwe do wykonania na specjalizowanych urządzeniach laboratoryjnych (analizy widm, szumów etc.), ale proste i skuteczne algorytmy wdrożone na poziomie kontrolerów maszyn. Łącząc tego typu mechanizmy z możliwością podłączenia dużej ilości urządzeń do sieci, uzyskujemy efekt skali, który przy określonym progu ilości maszyn do obsługi, daje wymierne korzyści związane z oszczędnością czasu pracy służb utrzymania ruchu.

Zaletą predykcji „przez sieć”, dla brygadzysty i inżynierów UR, jest scentralizowanie wszystkiego w jednym miejscu. Pracownicy techniczni nie muszą każdorazowo podchodzić do opomiarowanych maszyn, bo na bieżąco widzą ich status i wskazania zainstalowanych czujników (oczywiście pamiętać należy, że mówimy cały czas o pomiarach predykcyjnych, nie zwalnia nas to z obowiązku okresowych oględzin i działań prewencyjnych).

Wdrożenie połączenia maszyn w sieć, wprowadza do predykcji nowe możliwości, znacznego jej rozszerzenia i prostego dublowania mechanizmów, opartych o warstwę programową (np. w PLC albo systemach HMI). Dużo łatwiej jest skopiować i zaadoptować działający algorytm do nowego urządzenia, niż osobno doposażać je w samodzielny analizator sygnału czujników.

Podjęcie, przedstawione na przykładzie firmy Pilkington Automotive Poland pokazuje nam metodę zastosowania mechanizmów predykcyjnych

w taki sposób, aby były możliwie jak najlepiej zrozumiane i efektywnie wykorzystane przez pracowników na poziomie brygady reakcyjnej. Wizualizacja pomiarów z dużego obszaru przez sieć przemysłową pozwala na bieżąco analizować trendy pomiarów i podejmować szybkie decyzje z odpowiednim wyprzedzeniem czasowym. Skutkuje to wzrostem efektywności i skuteczności pracy działu utrzymania ruchu.

Przechodząc do ogółu, obecny poziom monitoringu stanu urządzeń w czasie rzeczywistym (stosowany w większości przedsiębiorstw), pozwala na utrzymanie pewnego ograniczonego poziomu niezawodności jako określania prawdopodobieństwa wystąpienia zdarzenia awaryjnego. Aby dążyć do doskonalenia technik predykcyjnych należy rozpocząć proces adaptacji mechanizmów sztucznej inteligencji i uczenia maszynowego. Pozwolą one na rozpoznawanie zaawansowanych zależności i zagnieżdżeń, które mogą zostać pominięte przez „ludzkie oko” [8].

Z punktu widzenia metodologii badań, za jakiś czas, należy przeanalizować wpływ adaptacji nowych technologii na firmy i organizacje związane z predykcyjnym utrzymaniem ruchu. Weryfikacji będzie należało poddać to, jak ludzie zintegrują się z nową generacją urządzeń automatyki i na jakim poziomie zaczną korzystać z algorytmów wykorzystujących sztuczną inteligencję w swoich organizacjach.

1.5. Podsumowanie

Podsumowując niniejszy artykuł, przedstawiono w nim te techniki predykcyjne, stosowane w dziale utrzymania ruchu, które wykorzystują mechanizmy przemysłowych sieci komunikacyjnych i szeroko pojętego Industry 4.0. Mamy więc tutaj wskazania z czujników, algorytmy zagnieżdżone w sterownikach PLC, informacje przesyłane przez sieć i gromadzone na centralizowanym serwerze, który oprócz wizualizacji pozwala na prezentację danych w formie raportów. W tych raportach, można zestawiać ze sobą wiele, pozornie niezwiązanych zmiennych i na ich podstawie wnioskować, o zależnościach mających wpływ na degradację stanu maszyn i urządzeń. Wykorzystanie mechanizmów sieciowych i udostępniania informacji pozwoliło na upowszechnienie wiedzy predykcyjnej i dostęp do wyników analiz już na poziomie pracowników wykonawczych. To powoduje, że wiedza, dotychczas zarezerwowana dla wąskiego grona specjalistów, zostaje przekazana „w dół”, a co za tym idzie, kompetencje całego zespołu idą „w górę”.

Przedstawiona w pierwszej części naszego artykułu ilość sterowników PLC czy robotów przemysłowych, w odniesieniu do powierzchni zakładu i długości linii produkcyjnych, może dać pewien obraz co do poziomu automatyzacji fabryki, w której można wdrażać sieciowe podejście do predykcji. Natomiast kluczową, jest kwestia poziomu „usieciowienia” zakładu i jak tę infrastrukturę mogą wykorzystać służby UR. Dlatego celem opracowania było pokazanie studium przypadku firmy Pilkington Automotive Poland, jako poradnika jak przygotować infrastrukturę i potem jakich metod używać, aby elastycznie zarządzać konserwacjami i planowaniem prac utrzymania ruchu.

Bibliografia

- [1] Biehl Wiedemann. *Website: Building Automation with AS-I*.
- [2] A. Blachowicz. Black Belt project: Decrease level of Technical Downtime for APBL line in Chmielów plant, February 2016.
- [3] Bourton Group. *Six Sigma Training Materials*, 2015.
- [4] Z.A. Bukhsh, I. Stipanovic, A. Saeed, A.G. Doree. Maintenance intervention predictions using entity-embedding neural networks. *Automation in Construction*, 116:103202, 2020.
- [5] Festo. *Festo Maintenance Tool - software documentation*.
- [6] Festo. *Installing and commissioning CPX terminals*, 2005.
- [7] L. Huang, M. Chen, Y. Liu. Learning-Aided Stochastic Network Optimization With State Prediction. *IEEE/ACM Transactions on Networking*, 26:1810–1820, August 2018.
- [8] V.J. Jimenez, N. Bouhmala, A.H. Gausdal. Developing a predictive maintenance model for vessel machinery. *Journal of Ocean Engineering and Science*, 2020.
- [9] I.-S. Jung, B.M. Mulman, D. Thapa, L.-J. Koo, J.-H. Bae, S.-H. Hong, S. Yeo, C.M. Park, S.C. Park, G.-N. Wang. PLC Control Logic Error Monitoring and Prediction Using Neural Network. *International Conference on Natural Computation*, 2:484–488, October 2008.

- [10] R. Keith-Mobley. *An Introduction to Predictive Maintenance*. Butterworth-Heinemann, 2002.
- [11] Movida. *Materialy konferencyjne: XI FORUM Służb Utrzymania Ruchu*, 2017.
- [12] M. Rafeeq, A. Afzal, S. Rajendra. Remote Supervision and Control of Air Conditioning Systems in Different Modes. *J. Inst. Eng. India Ser. C*, strony 100–175, 2019.
- [13] Rockwell Automation. *Publication ENET-UM527E-EN-P*, 2006.
- [14] Rockwell Automation. *Publication FTALK-PP028I-EN-P*, 2016.
- [15] Rockwell Automation. *Publication HSE-IN025D-EN-E*, 2018.
- [16] Rockwell Automation. *Publication HSELD-UM024C-EN-E*, 2018.
- [17] A. Syaichu-Rohman, R. Sirius. Model predictive control implementation on a programmable logic controller for DC motor speed control. *Proceedings of the 2011 International Conference on Electrical Engineering and Informatics, Bandung.*, 1:1–4, 2011.
- [18] R. K. Valencia-Palomo, G. and Hilton, J. A. Rossiter. Predictive Control implementation in a PLC using the IEC 1131.3 programming standard. *2009 European Control Conference (ECC), Budapest.*, strony 1317–1322, 2009.

Machine breakdowns prediction, using industrial networks and industry 4.0 mechanism

Abstract: Modern production plants are inevitable coming to achieve standards determined by fourth industry revolution. Together with change in machines design, also change technical departments (especially Maintenance), management and staff approach to diagnostic and breakdown prediction process. In those plants, especially working for automotive industry, architecture of machines control and their metering allows us to implement automatic prediction mechanism. Programmable logic controllers network connection, give us access to process and machines data, which we can use to prepare prediction algorithms. Moreover, modern PLCs allows us to create advanced subroutines to implement prediction at the level of single machine. Collected data can be visualised directly on machine operation panel, and stored in central database for reporting portal or MES system. Those features of industry network connection, allows us to create system of supervision and management of the machine park.

2. Budowa zdecentralizowanej sieci czujników bezprzewodowych na potrzeby akwizycji danych pomiarowych z linii produkcyjnych

LESZEK KLICH¹

POLITECHNIKA RZESZOWSKA, L.KLICH@PRZ.EDU.PL

Streszczenie Bezprzewodowe sieci sensorowe (ang. Wireless Sensor Networks) znajdują zastosowanie zarówno w przemyśle, jak i w gospodarstwach domowych („cyfrowy dom”), w zakresie akwizycji danych z wielu punktów pomiarowych. Typowe sieci sensorowe budowane są w topologii gwiazdy, z centralnym punktem zarządzania, będącym jednocześnie punktem akwizycji danych. Publikacja opisuje budowę sieci bezprzewodowych czujników, przy wykorzystaniu transceiverów pracujących na częstotliwości 433MHz. Zbudowana sieć jest odporna na awarie dzięki eliminacji punktu centralnego oraz możliwości wykorzystania każdego z węzłów jako węzła pośredniego. Ponadto, każdy z węzłów może być wykorzystany jednocześnie jako węzeł pośredni, dzięki dzięki algorytmowi dynamicznej tablicy z zapisem danych z innych węzłów, każdy węzeł może być użyty jako centralny punkt akwizycyjny. Dodatkowo, sieć pozwala na analizę awarii jednego lub wielu węzłów w sieci.

¹ORCID: 0000-0001-6099-2417, Wydział Mechaniczno-Technologiczny Politechniki Rzeszowskiej, Kwiatkowskiego 4, 37-450 Stalowa Wola

2.1. Wprowadzenie

Współcześnie stale rośnie obszar zastosowania oraz moc obliczeniowa systemów przetwarzających dane. Z kolei postęp techniczny i malejące ceny mikrokontrolerów oraz komponentów implikuje wzrost zapotrzebowania na wszelkiego rodzaju sensory, mogące zasilić programy kontrolerów w niezbędne do analizy dane. Różnego rodzaju czujniki udostępniające dane w formie analogowej czy cyfrowej, wykorzystywane są od wielu lat i znajdują zastosowanie, zarówno w rozwiązaniach profesjonalnych, jak i amatorskich. Nowoczesne sensory udostępniają wyniki w formie cyfrowej, a dzięki temu zapewniają większą dokładność oraz niezawodność pomiarów.

Ze względu na coraz większą dostępność różnego rodzaju sensorów pomiarowych, wykorzystujących różnego rodzaju interfejsy, osiągnięcie coraz większej dokładności pomiarów staje się coraz łatwiejszym zadaniem. Projektant systemu pomiarowego musi zadbać natomiast o zapewnienie niezawodnych i często redundantnych kanałów transmisyjnych, które zapewnią wysoką niezawodność akwizycji danych [10].

Od lat zainteresowanie badaczy skupione jest na technologiach bezprzewodowych. Istnieje bowiem w tym obszarze wiele technologii, które umożliwiają jedno lub dwukierunkową komunikację na coraz większe odległości, przy jednoczesnym zachowaniu rozsądnego poboru energii. Co interesujące, wykorzystanie współczesnych technologii bezprzewodowych umożliwia budowę sieci rozległych, poprzez które podłączyć można praktycznie nieograniczoną liczbę czujników do urządzenia centralnego. Na potrzeby zwiększenia dystansu i niezawodności oraz zmniejszenia energii w tego typu sieciach, powstają nowe implementacje protokołów routingu. [4]

Jedną z niewątpliwych zalet technologii bezprzewodowej, jest duża elastyczność jej organizowania w tzw. węzły, które odpowiednio zaprogramowane mogą adaptować trasę pakietów, dzięki identyfikacji innych węzłów oraz ilości przeskoków, tworząc optymalne struktury. Co więcej – dane w jednym węźle mogą automatycznie retransmitować dane do węzłów sąsiednich, a te z kolei do kolejnych [18].

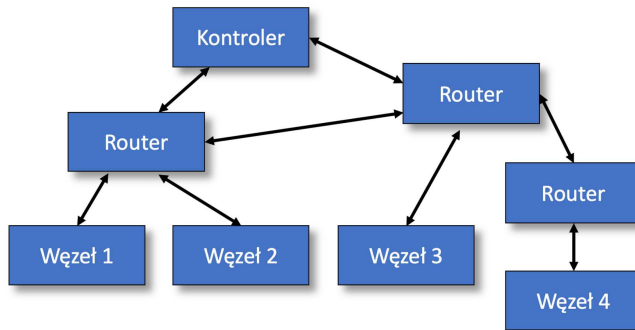
Systemy bezprzewodowych sieci czujników stosować można także na duże dystansy w rozległych sieciach, dzięki wykorzystaniu topologii drzewiastej, która dzięki algorytmom routingu umożliwia transmisję danych na bardzo duże odległości przy wykorzystaniu mechanizmu przeskoków [9].

Celem opisanego w niniejszej pracy projektu było zbudowanie bezprzewodowej akwizycji danych, wykorzystującej zabezpieczony tor transmisyjny przy wykorzystaniu ogólnie dostępnych transceiverów, działających na częstotliwości 433 Mhz i połączonych w sieć. Założeniem do projektowania systemu było uzyskanie rozproszonej sieci czujników bezprzewodowych, pracujących w topologii kratowej i przesyłających przykładowe dane pomiarowe. Opisano także wyniki badań z możliwości zbudowania tego typu sieci czujników, w której dodanie bądź uszkodzenie jednego lub n czujników spowoduje adaptację pozostałych węzłów do nowej liczby czujników i ich dalszą nieprzerwaną pracę na pozostałych węzłach. Przedstawiono także propozycję rozwiązania problemów wynikających z braku kontrolera (punktu centralnego) w szczególności występujących w tego typu sieciach zakleszczeń. Każdy z czujników posiada możliwość zasilania z portu USB, jak również wyposażony jest w akumulator, który umożliwia pracę bez podłączonego zasilania. Moduły mogą komunikować się z komputerem poprzez port USB, przesyłając dane w formacie JSON, zatem każdy moduł może zostać wykorzystany, jako odrębny akwizytor danych ze wszystkich czujników. Oprócz tego, projekt zawiera osobny moduł z wyświetlaczem LCD, który posłużył do testów zasięgu sieci. Dodatkowo zaprojektowano multiplatformowe oprogramowanie GUI do akwizycji czujników, które współpracuje z dowolnym modułem sieci oraz akwizytorem danych, przetwarzając format JSON na CSV, który z kolei można analizować za pomocą dowolnego oprogramowania. Podczas eksperymentu zbadane zostało działanie oprogramowania oraz przewidywany zasięg sensorów.

2.2. Najważniejsze pojęcia

Bezprzewodowe sieci sensorowe to sieci, które składają się z rozproszonych, autonomicznych, których głównym zadaniem jest monitorowanie określonych parametrów (sensory). Urządzenia te mogą zawierać urządzenie pomiarowe oraz nadajnik lub transceiver radiowy. Dane pomiarowe wysyłane są drogą radiową do punktów centralnych, gdzie następuje ich właściwa obróbka. Zastosowanie bezprzewodowych sieci sensorowych jest uniwersalne, bowiem mierzyć one mogą dowolne wartości fizyczne na rozległym obszarze. Często dobrym rozwiązaniem jest tworzenie własnych aplikacji – dedykowanych do konkretnych wymagań, co pozwala osiągnąć żądany kompromis między sprzecznymi parametrami (np. energooszczędność i duża przepływność), niedostępny w standardowych rozwiązaniach [19].

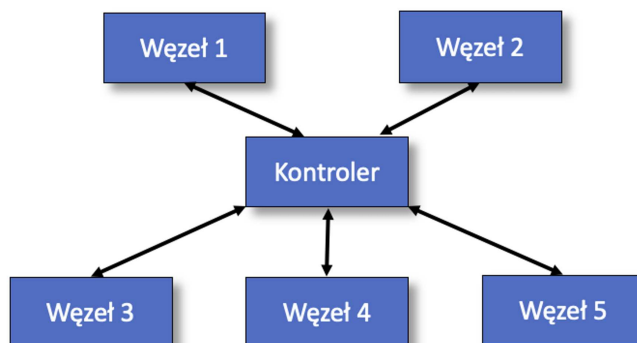
Topologia kratowa (mesh) wykorzystywana jest najczęściej w aplikacjach militarnych, w których kluczowa jest niezawodność czy zdolność do funkcjonowania w skrajnie trudnych warunkach. Dane w topologii kratowej przekazywane są pomiędzy pojedynczymi węzłami, dopóki nie dotrą do przeznaczenia. Warto podkreślić, że tego typu sieci nie muszą być wyposażone w punkt centralny, którym w sieciach bezprzewodowej akwizycji czujników pomiarowych często są bramki sterujące pracą sieci. Zamiast tego, węzły są równorzędne względem siebie i komunikują się z innymi węzłami w obrębie zasięgu radiowego. Dzięki temu, nie ma znaczenia zmiana położenia węzła lub utracenie zasięgu przez węzeł, bowiem występuje dynamiczne trasowanie. Topologia mesh realizowana jest w systemach komputerowych, przy wykorzystaniu sieci komputerowych oraz dzięki wykorzystaniu dodatkowych sterowników, np. MCL (Mesh Connectivity Layer) lub innych sterowników dostępnych w systemach operacyjnych lub w oprogramowaniu routerów [15].



Rysunek 2.1: Schemat sieci czujników bezprzewodowych w topologii kratowej.

Topologia gwiazdy składa się z koordynatora (bramki) i urządzeń końcowych (węzłów). W topologii tej, urządzenie końcowe komunikuje się wyłącznie z koordynatorem i wymiana pakietów między urządzeniami końcowymi musi przebiegać przez koordynatora. Wadą topologii jest działanie sieci zależne od sprawności koordynatora, co powoduje, że koordynator staje się tzw. wąskim gardłem. Inną wadą jest brak alternatywnej ścieżki od źródła do miejsca docelowego. Zaletą zaś jej prostota oraz niewielka ilość przeskoków, wynosząca co najwyżej dwa, by dotrzeć do miejsca docelowego.

Topologia drzewa to rodzaj sieci składającej się z centralnego węzła

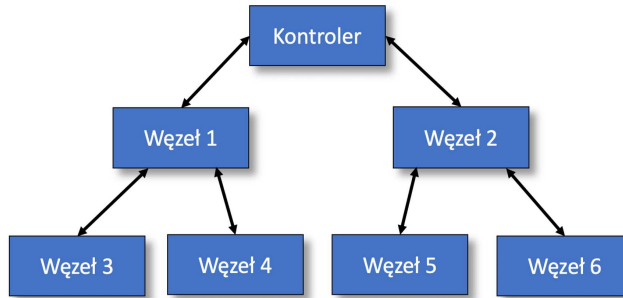


Rysunek 2.2: Schemat sieci czujników bezprzewodowych w topologii gwiazdy.

(drzewa głównego), który jest koordynatorem oraz kilku routerów i urządzeń końcowych, jak pokazano na Rysunku 2.3. Funkcją routera jest zwiększenie zasięgu sieci. Węzły końcowe, które są podłączone do koordynatora lub routerów, nazywane są „dziećmi”. Tylko routery i koordynator mogą mieć dzieci. Każde urządzenie końcowe może komunikować się tylko z rodzicem (routerem lub koordynatorem). Koordynator i routery mogą mieć dzieci i dlatego są jedynymi urządzeniami, które mogą być rodzicami. Urządzenie końcowe nie może mieć dzieci i dlatego nie może być rodzicem. Szczególnym przypadkiem topologii drzewa jest topologia drzewa klastrowego. Topologia drzewa posiada zasadniczą wadę, polegającą na zależności węzłów końcowych od rozgałęzień „rodziców”. W przypadku, gdy zostanie ono wyłączone, dzieci wyłączonego rodzica nie mogą komunikować się z innymi urządzeniami w sieci. W topologii drzewa dwa odgałęzienia, nawet będące blisko siebie, nie mogą komunikować się bezpośrednio. Topologia drzewa umożliwia wykorzystanie technologii routera, lecz nie umożliwia zmiany ścieżki dotarcia pakietów.

Dynamika systemu pomiarowego wiąże się z ilością cykli pomiaru w jednostce czasu, czyli częstotliwością próbkowania. W warunkach eksperymentu, gdzie jako medium transmisji wykorzystano fale radiowe o częstotliwości 433 MHz, dynamika pomiaru może niekorzystnie rosnąć, przez niekorzystny wpływ wielu czynników, takich jak transmisja szeregową oraz niewielka szybkość łącza wynosząca około 2 Kb/s przy ramce pakietu o długości maksymalnej pamięci podręcznej. Zatem czas transmisji T_d wynosi:

$$[h]T_d = \frac{\text{bitów (w pakiecie)}}{\text{szybkość transmisji}} \quad (2.1)$$



Rysunek 2.3: Schemat sieci czujników bezprzewodowych w topologii drzewa.

a zatem liczba próbek w ciągu sekundy L teoretycznie wynosić może:

$$[h]L = 1/T_d \quad (2.2)$$

W przypadku systemów bezprzewodowych uwzględnić także należy czas propagacji fali radiowej T_p , który jest dłuższy od T_d , jednakże należy doliczyć także czas przełączania transceivera w tryb transmit/receive oraz czas oczekiwania na inne urządzenia. W konsekwencji, im więcej transceiverów w systemie pomiarowym, tym mniejsza dynamika systemu. Wynika to z konstrukcji samych modułów transceiverów oraz stosunkowo niskiej częstotliwości pracy. Należy także uwzględnić zakłócenia, które w paśmie 433 Mhz są powszechne, zatem w praktyce należy stosować inne zależności do obliczania dynamiki pomiaru [14].

Numer sieci to liczba identyfikująca sieć, w której znajduje się n węzłów, gdzie n to liczba z przedziału 1-255. Numer sieci powinien być identyczny we wszystkich urządzeniach pracujących w danej sieci, by możliwa była ich wzajemna komunikacja. Sieć o określonym numerze jest jednocześnie odizolowana od innych, zatem moduły w sieci 1 nie będą komunikować się z sieciami o innym numerze. Maksymalna ilość sieci wynosi 256.

Adresowanie to sposób identyfikacji urządzeń w sieci. Każde urządzenie (czujnik) musi posiadać swój adres. Adres składa się z dwóch liczb numeru sieci oraz unikalnego numeru węzła w sieci. Każda z liczb zawiera się w zakresie 1-255. Dla przykładu adres składa się z numeru sieci oraz numeru urządzenia.

Tryb promiscuous to tryb odbierania danych od wszystkich. Wiadomości przesyłane między węzłami są prywatne, zatem węzły ignorują wiadomości, które nie są do nich skierowane. Jednakże można włączyć specjalny tryb, który spowoduje, że węzeł otrzyma każdą wiadomość, która pojawia się w danej sieci. Jest to szczególnie użyteczny tryb w przypadku mechanizmu routingu lub debugowania. Należy jednak pamiętać, że tryb ten dotyczy wyłącznie pojedynczej sieci.

Acknowledgement to potwierdzenie odbioru. Wysyłając dane do innego urządzenia, można wybrać jeden z dwóch trybów – tryb bez potwierdzenia odebrania, lub tryb z potwierdzeniem. Potwierdzenie to zwykle krótki komunikat „ACK” oznaczający, że pakiet został poprawnie odebrany. Biblioteka dostarczona do transceivera RFM69, które jest nadajnikiem/odbiornikiem użytym do eksperymentu pozwala na automatyczne ponowienie wysyłki w przypadku, gdy nie otrzyma potwierdzenia z urządzenia docelowego, ale może także ponownie przesłać dane bez tego typu odpowiedzi. Użycie trybu zależy bowiem od typu jego zaprogramowania. Niektóre aplikacje mogą z powodzeniem działać bez tego trybu. Warto także pamiętać, że żądanie potwierdzenia odebrania wiadomości wiąże się z koniecznością sprawdzenia statusu potwierdzenia, a czasem „ręcznego” wysłania wiadomości „ACK”.

Encryption to szyfrowanie przesyłanych danych. W przypadku RFM69 jest to AES (Advanced Encryption Standard). Szyfrowanie kluczem symetrycznym uniemożliwia podsłuchanie transmisji nawet, jeśli atakujący posiada numer sieci, zaś przesyłane pakiety nadawane są w trybie promiscuous. Szyfrowanie włącza się za pomocą biblioteki, poprzez zdefiniowanie we wszystkich urządzeniach identycznego 16-bajtowego klucza szyfrującego.

SPI to szeregowy interfejs (Serial Peripheral Interface) służący do dwukierunkowej (full-duplex), synchronicznej transmisji danych pomiędzy mikrokontrolerem a układami peryferyjnymi. Interfejs jest trójprzewodowy i składa się z dwóch linii przesyłających synchronicznie dane w przeciwnych kierunkach oraz linii zawierającej sygnał taktujący, którego zadaniem jest synchronizacja transferu danych.

2.3. Założenia i model działania

Konieczność bezprzewodowej akwizycji danych, nieodzowna jest zarówno w obiektach przemysłowych, jak i innych zastosowaniach, gdzie niezbędne są pomiary w różnych miejscach, za pomocą sieci czujników, które przesyłają dane do centralnego punktu. W skrajnych przypadkach przeprowadzenie przewodów, np. ze względu na części ruchome bądź ograniczoną długość przewodu może być utrudnione lub wręcz niemożliwe. W takich przypadkach celowe staje się zastosowanie rozwiązań bezprzewodowych. Należy jednak pamiętać, że medium przesyłu, jakim jest powietrze, narażone jest na wiele negatywnych czynników, które należy uwzględnić przy ich projektowaniu. Zasięg nie jest tu jedynym kryterium, choć istotne jest wdrożenie algorytmu kontrolującego jakość sygnału i analizę dostępności innych czujników w sieci. Transmisja bezprzewodowa wiąże się także wydłużeniem czasu transmisji czy awariami poszczególnych urządzeń [8].

W niniejszym opracowaniu rozważany jest nietypowy model sieci, składający się wyłącznie z węzłów czujnikowych, pełniących jednocześnie funkcje: czujnika, przekaźnika oraz punktu końcowego akwizycji danych dla współpracy z komputerem. Każdy węzeł jest wyposażony w algorytm magazynowania danych ze wszystkich węzłów w sieci. Każdy z węzłów w sieci jest również zasilany z akumulatora i okresowo identyfikuje dostępność sieci (wolne pasmo), a także wysyła tablicę danych do wszystkich sąsiadujących węzłów w sieci. Każdy węzeł posiada także własny identyfikator. Brak punktu centralnego powoduje, że węzły uczestniczą w przesyłaniu danych z innych węzłów, dzięki włączeniu rozgłaszania nieadresowanego, co powoduje, że każdy pakiet wysłany z jednego węzła trafia do wszystkich węzłów będących w zasięgu jednego węzła. Konstrukcja tego typu routingu opiera się na założeniu, że przynajmniej dwa sąsiednie węzły wymieniają ze sobą dane, zaś niektóre z nich nie mogą odbierać pakietów ze wszystkich węzłów. Mimo to, otrzymują one dane z tablicy danych sąsiednich węzłów, które przechowują dane odebrane z węzłów spoza zasięgu.

Na potrzeby projektu przyjęto następujące założenia:

1. W systemie może pracować może wiele równorzędnych modułów.
2. Każdy z modułów pracuje niezależnie lub jeśli istnieje więcej czujników, wymienia z nimi dane.
3. Sieć czujników pozbawiona jest punktu centralnego (sterującego).

4. Zastosowane elementy (mikrokontroler i transceiver) są możliwie tanie i łatwe w produkcji.
5. Moduły i ich firmware mogą być w prosty sposób modyfikowane i rozbudowywane.
6. Moduły mogą być rozmieszczone obok siebie (redundancja) lub rozmieszczone po pewnym obszarze. W drugim przypadku urządzenia przekazują dane do pozostałych urządzeń, nawet wtedy, gdy dwa skrajnie umieszczone moduły nie mają ze sobą zasięgu.

Na podstawie założeń zaprojektowano i zbudowano prototyp urządzeń – czujników oraz opracowano firmware na potrzeby przeprowadzenia testów.

Algorytm działania sieci jest następujący:

Pojedynczy węzeł czeka losowy czas na odbiór pakietu w sieci. Jeśli w tym czasie otrzyma pakiet danych, buduje dynamiczną strukturę zawierającą odebrane dane oraz dołącza dane z własnego czujnika. Następnie losuje czas oczekiwania i jeśli nie odbierze danych z sieci, wysyła pakiety zawierające dane z tablicy dynamicznej. Jednocześnie podczas każdej pętli oczekiwania, zwiększany jest timeout każdego odebranego rekordu danych. Jeśli pojawiają się dane z dowolnego czujnika, porównywany jest timeout. W sytuacji, gdy timeout jest mniejszy od istniejącego w dynamicznej strukturze, rekord podmieniany jest rekord zawierający dane z konkretnego czujnika i jednocześnie zerowany timeout. Tym samym, w kolejnym nadawaniu, wysyłane są najświeższe dane, które zdobył węzeł. W przypadku, gdy brak jest dostępnych pakietów w sieci, węzeł transmituje własne tablice danych oraz stale zwiększa timeout w każdym z rekordów tablicy. Przekroczenie tego parametru o ustaloną liczbę, powoduje usunięcie danych z tablicy podręcznej. Ostatecznie po wygaśnięciu czasu życia pakietów (przekroczony timeout), węzeł nadaje w losowym czasie wyłącznie pakiet zawierający dane z własnego czujnika. Dzięki mechanizmowi timeout, który jest wspólny we wszystkich węzłach, każdy z nich może wnioskować na podstawie granicznej wartości o błędzie, uszkodzeniu lub wyłączeniu jednego lub n czujników i podjąć określone decyzje, które jednak na tym etapie prac nie zostały jeszcze zaimplementowane w kodzie programu węzłów.

Struktura systemu sensorów opiera się na samodzielnie wykonanych modułach czujników, zbudowanych na bazie mikrokontrolera Atmega 328, zaś każdy z nich podłączony jest do transceivera pracującego na częstotliwości

433 Mhz. Ze względu na brak centralnego arbitra (niezależnego kontrolera centralnie zarządzającego wszystkimi sensorami w sieci), praca czujników odbywa się na podstawie identyfikacji wolnego pasma i losowych opóźnień. Każdy z modułów w sieci w trybie nasłuchu odbiera dane ze wszystkich urządzeń. Zebrane dane przechowywane są w dynamicznie modyfikowanej na bieżąco strukturze, w zależności od wartości czasowych odebranych z innych urządzeń.

Warstwa sieciowa została zorganizowana w topologii kratowej (mesh). Główną przewagą tego typu sieci, to zdolność dynamicznej samoorganizacji i autokonfiguracji oraz zdolność do automatycznej łączności między węzłami w sieci, podczas gdy np. w konwencjonalnej sieci o topologii gwiazdy wszystkie węzły końcowe są połączone z jednym centralnym punktem, będącym górnym poziomem sieci [11].

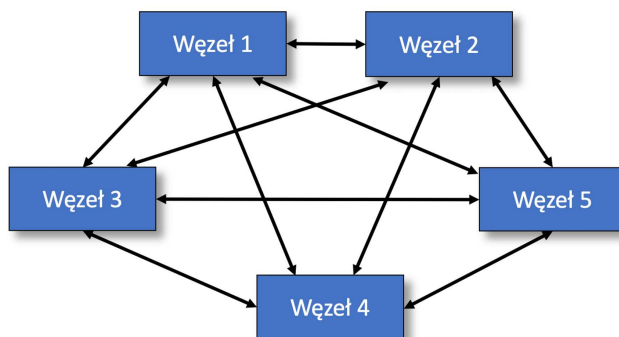
W układzie testowym zastosowano 5 czujników oraz jedno urządzenie wyposażone w wyświetlacz LCD 16x2 oraz przyciski, który umożliwia odczyt danych z dostępnych modułów oraz przełączanie wyświetlania. Moduły wymieniają dane pomiędzy sobą, zaś dodanie lub odłączenie jednego lub wielu modułów nie spowoduje awarii sieci. W przypadku wykrycia kolizji, moduły wstrzymują działanie i każdy moduł generuje liczbę pseudolosową, będącą opóźnieniem startu ponownej transmisji. Odebranie pakietu danych przez inne urządzenia oznacza ich przejście w tryb odbioru [1].

W sieciach typu „wireless sensor network” moduły pomiarowe kontrolowane są przez dodatkowe urządzenia pełniące funkcję bramy i/lub kontrolera sieci. Dzięki temu czujniki wysyłają dane jedynie do jednego punktu (gateway). Dodatkowo moduły mogą pełnić funkcję repeatera lub routera, co umożliwia przekazywanie i odbieranie danych z innego węzła czujników, przekazując je z powrotem do bramy (Rysunek 2.2).

Poszczególne węzły mogą także rozwidlać się, dając możliwość utworzenia topologii drzewa, w którym stosuje się adresowanie czujników oraz możliwe jest tworzenie podsieci. Tego typu rozwiązanie jest logiczne w przypadku zastosowanego medium transmisyjnego, czyli częstotliwości radiowej, o niewielkiej szybkości transmisji danych. Istotną zaletą kontroli sieci przez punkt centralny jest możliwość wprowadzenia mechanizmu potwierdzania odbioru transmisji (ACK), co znacząco zwiększa niezawodność akwizycji. Na tej zasadzie, w uproszczeniu działają technologie LoRa oraz ZigBee.

W praktyce występują także sieci o strukturze drzewa klastrowego, zawierające koordynatora sieci, routery oraz urządzenia końcowe [6].

Rezygnacja z bramy kontrolującej sieć powoduje, że czujniki znajdujące się w jednej sieci pozbawione są koordynatora. W konsekwencji sieć tego typu pozbawiona jest arbitra, który kontroluje system czujników. Niższe opracowanie poświęcono zbadaniu możliwości zaprojektowania sieci czujników bezprzewodowych, pozbawionych punktu centralnego i zbadanie możliwości pracy tego typu sieci. Wyzwaniem przy budowie takiej sieci są różnorakie problemy jakie mogą się pojawiać w trakcie, np. zaprogramowanie zaprogramowanie algorytmów detekcji ilości sąsiednich czujników, identyfikacja wolnego pasma i wyzwalanie funkcji nadawania oraz detekcja i rozwiązywanie kolizji. Dodatkowo, tego typu sieć posiada dynamicznie zmieniającą się strukturę (ilość węzłów oraz zasięg), co oznacza, że topologia może dynamicznie się zmieniać, w zależności od wzajemnego rozłożenia i przeszkód terenowych. Aby uprościć protokół transmisji, zrezygnowano mechanizmu potwierdzania odbioru, zdając się na duże prawdopodobieństwo dotarcia pakietu do jednego z wielu węzłów, które prześlą dane do czujników będących w ich zasięgu (Rysunek 2.4).



Rysunek 2.4: Wizualizacja przepływu danych pomiędzy węzłami w eksperymentalnej sieci bezprzewodowej pozbawionej punktu centralnego.

Początkowym założeniem było wdrożenie jednego z dostępnych mechanizmów routingu dynamicznego, aby urządzenia (węzły) mogły dynamicznie przekazywać pakiety przy uwzględnieniu najkrótszej drogi – podobnie jak czynią to tradycyjne routery [5]. Jednakże już początkowy etap badań wykazał, że istnieją istotne ograniczenia sprzętowe, wynikające z ni-

skiej prędkości transmisji, niewielkiej ilości pamięci RAM (2 KB) dostępnej w zastosowanych mikrokontrolerach, niskiej częstotliwości zegara oraz zbyt małej ilości pamięci FIFO w module transceivera. Należało zatem przemodelować koncepcję wymiany danych pomiędzy czujnikami. Rozwiązanie problemu polegało na zastosowaniu statycznego routingu, jednakże wymagało to zmian konfiguracji węzłów za każdym razem, gdy zmieni się położenie czujników (obliczanie odległości na podstawie siły sygnału RSSI). W związku z tym, nie gwarantowało dotarcia pakietu najkrótszą drogą do miejsca przeznaczenia, w przypadku zmiany rozmieszczenia czujników bez rekonfiguracji sieci.

Z uwagi na powyższe, zmieniono koncepcję na sieć równoważnych węzłów, w których każdy pełni podwójną rolę – urządzenia wysyłającego dane z podłączonego czujnika oraz przechowującego i wysyłającego dane o innych węzłach. Każdy z węzłów wyposażony jest w dynamiczną listę struktur, w której urządzenie przechowuje: id czujnika, parametr odczytany z czujnika lub innych urządzeń oraz znacznik czasu – przekazany z innego czujnika oraz inkrementowany od czasu odebrania danych. Urządzenie aktualizuje informację w rekordzie struktury, informacją pobraną z własnego czujnika. Następnie wysyła poszczególne rekordy struktury z innych czujników, po czym przechodzi w tryb nasłuchu. Odbierając dane z każdego z urządzeń, każde urządzenie analizuje odebrane dane i porównuje z wewnętrzną strukturą danych. W przypadku przekroczenia zmiennej *timestamp*, węzeł jest usuwany ze struktury, co oznacza, że czujnik został wyłączony lub uszkodzony. Co istotne, zanim nastąpi usunięcie węzła, na podstawie znacznika czasu wnioskować można, o jego możliwym uszkodzeniu.

Wybór transceiverów był wyzwaniem, bowiem zostały one przeznaczone głównie do pracy *peer-to-peer* lub w topologii gwiazdy z dodatkowym kontrolerem centralnym, koordynującym pracę sieci. Celowo zrezygnowano z dostępnych technologii typu LoRa czy ZegBee, aby zbadać możliwość wykorzystania popularnych i tanich transceiverów dostępnych na rynku.

Dla uproszczenia procesu badawczego, każdy z sensorów zawiera jedynie czujnik temperatury. Jednakże nic nie stoi na przeszkodzie, aby urządzenia końcowe wyposażone było w inne czujniki, bądź większą ich ilość, aktywowanych w zależności od potrzeby.

2.4. Transceiver RFM69HW

Moduł transceivera RFM69 (Rysunek 2.5) występuje w kilku wersjach. W systemie testowym zostały zastosowane moduły pracujące na częstotliwości 433 MHz. Cechą charakterystyczną modułów jest możliwość ich programowej konfiguracji. Charakteryzują się one niskim poborem prądu w trybie uśpienia, który wynosi nawet $0.5 \mu\text{A}$, zaś w trybie odbioru pobór prądu wynosi 16mA. Pobór prądu podczas nadawania, uzależniony jest od mocy nadawanego sygnału. Minimalne napięcie zasilania dla trybu RX wynosi 2.4 V.



Rysunek 2.5: Moduł RFM69 - wyprowadzenia sygnałów modułu.

Moduł RFM69 pracuje z logiką 3.3 V, zatem należało uwzględnić to przy wyborze mikrokontrolera. W przypadku wykorzystania mikrokontrolerów pracujących z logiką 5 V, należy wyposażyć układ w konwerter napięć logicznych. Moduł cechuje szeroki zakres temperatury pracy, wzmacnienie +20 dBm przy 100 mW prądu oraz czułość do -120 dBm przy prędkości 1.2 kbps.

Moduły RFM69 to zaawansowane transceivery przeznaczone do pracy z modulacją FSK, GFSK, MSK, GMSK i OOK. Transceiver posiada własne metody modulacji. Modulację należy rozumieć jako modyfikację sinusoidalnych fal radiowych. Przekazywanie danych za pomocą fali nośnej polega na modyfikacji parametrów tej fali poprzez nałożenie ich na falę nośną, aby dostarczyć dane do odbiornika i zdekodować z powrotem na dane. Proces ten nazywa się modulacją w nadajniku i demodulacją w odbiorniku (Rysunek 2.6).

Dodatkowo zostały one wyposażone moduł obsługi danych pakietowych

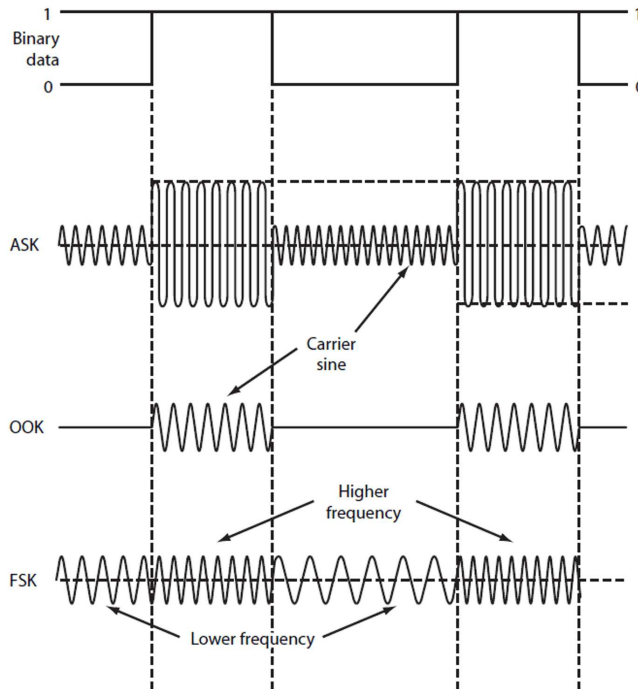
Tablica 2.1: Wyprowadzenia pinów modułu - nota katalogowa.

Oznaczenie	Nazwa	Opis
MISO	Master In Slave Out	Dane z modułu do mikrokontrolera
MOSI	Master Out Slave In	Dane z mikrokontrolera do modułu
SCK	Serial clock	Sygnal zegara z mikrokontrolera do modułu
NSS	Slave select	Sygnal select z mikrokontrolera do modułu
DIO0	Digital I/O 0 RX interrupt	Sygnal po odebraniu danych dla mikrokontrolera
AN	Antena	Podłączenie anteny
GND	Ground	Masa modułu
3.3 V	Power	Zasilanie układu 1.8-3.6 V (130 mA)

Tablica 2.2: Kluczowe parametry modułu RFM69.

Nazwa parametru	Wartość
Moc wyjściowa	+20 dBm - 100 mW
Czułość odbiornika	-120 dBm przy prędkości 1.2 kbps
Szybkość transmisji	do 300 kb/s przy modulacji FSK
Szyfrowanie	AES-128
Suma kontrolna	CRC-16
Pamięć podręczna	RX/TX FIFO 66 bajtów
Temperatura pracy	od -40°C do $+85^{\circ}\text{C}$
Dodatkowe	Wbudowany czujnik temperatury

z buforem FIFO, blokiem do obliczania sumy kontrolnej (CRC), a także możliwością szyfrowania danych w standardzie AES (tylko RFM69W). Moduł RFM69W występuje także w wersji C (RFM69CW, RFM69HCW), kompatybilnej pod względem wyprowadzeń z produktami starszych gene-



Rysunek 2.6: Rodzaje cyfrowej modulacji sygnału w module RFM69 w identyfikacji stanów logicznych (ASK) – modulacja amplitudowa, w której stany logiczne rozróżniane są na podstawie amplitudy (carrier sine), (OOK) – kluczkowanie on/off, (FSK) – kluczkowanie częstotliwości, w której identyfikacja stanów odbywa się na podstawie odchylenia częstotliwości (na schemacie oznaczone jako higher frequency).

racji oraz w wersji H (RFM69HW, RFM69HCW), odznaczającej się zwiększoną mocą nadajnika.

W urządzeniach prototypowych zastosowano modulację częstotliwości (FSK), która bazuje na zmianie współczynnika ułamkowego dzielnika w pętli sprzężenia zwrotnego PLL. Idea modulacji częstotliwości polega na kodowaniu informacji poprzez zmianę częstotliwości sygnału nośnego. Należy zauważyć, że modulacja FSK jest analogiczna do modulacji FM, z tą jednak różnicą, że w przypadku modulacji FSK, sygnał modulujący jest sygnałem binarnym.

Duża rozdzielczość modulatora sigma-delta w modułach RFM69 pozwala na bardzo wąskie odchylenie częstotliwości FDEV, określone przez:

$$[h]F_{DEV} = F_{STEP} \cdot F_{DEV} \quad (2.3)$$

Jednakże według noty katalogowej minimalne odchylenie modulacji dla prawidłowego połączenia jest ograniczone w zależności od prędkości transmisji (BR) do:

$$[h]F_{DEV} = \frac{BR}{2} \leq 500kHz \quad (2.4)$$

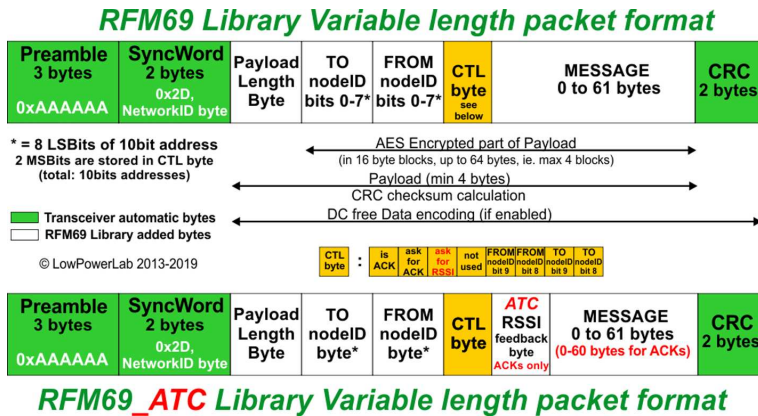
Choć moduł nie posiada ograniczenia wskaźnika odchylenia częstotliwości modulacji, nota katalogowa podaje, że odchylenie to musi być większe od 600Hz.

Tablica 2.3: Podłączenie RFM69 do mikrokontrolera.

Nazwa pinu	328 (Pro/Mini/Uno)	Mega
MISO	12 lub ICSP-1	50 lub ICSP-1
MOSI	11 lub ICSP-4	51 lub ICSP-4
SCK	13 lub ICSP-3	52 lub ICSP-3
NSS	10	53
DIO0	2	2
3.3 V	3.3 V VCC	3.3 V VCC
GND	Masa	Masa

Moduł RFM69, celem oszczędzania energii, nie przekazuje danych w sposób ciągły, lecz dzieli je na pakiety określonej wielkości (61 bajtów). Mogą one być jednak przekazywane wielokrotnie, dając wrażenie ciągłego połączenia. Format przesyłanego komunikatu zależy zaś od programu sterującego.

Do obsługi każdego modułu czujnika użyto bibliotek LowPowerLab [2] oraz biblioteki ArduinoJson (<https://github.com/bblanchon/ArduinoJson>) – do serializacji danych na port szeregowy. Struktura nagłówka pakietu RFM69 – w tym rozbiecie bajtu CTL wygląda następująco:



Rysunek 2.7: Struktura pakietu z podziałem na część dodawaną automatycznie przez transceiver (3 bajtowa preambuła, 2 bajty synchronizacyjne, konfiguracja transmisji CTL, suma kontrolna pakietu CRC) oraz bibliotekę obsługi (rozmiar, adres urządzenia docelowego i źródłowego, oraz komunikat).

Jak wynika ze schematu (Rysunek 2.7), transceiver może w jednym pakiecie przesłać maksymalnie 61 bajtów danych.

Biblioteka obsługi transceiverów RFM69 posiada dwa tryby działania. Jeden z nich przewiduje wskazanie miejsca przeznaczenia dotarcia pakietu, zatem w tym przypadku istnieje możliwość zaadresowania pakietu do wskazanego odbiorcy. Innym trybem jest adresowanie do wszystkich lub włączenie w module trybu *promiscuous*, który oznacza włączenie podsłuchu wszystkich pakietów w sieci. W tym wypadku niezależnie od tego, który moduł nadaje, wszystkie inne odbierają pakiety.

Listing 2.1: parametry adresacji modułów

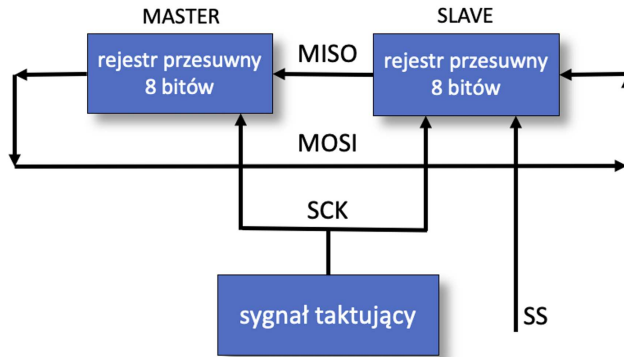
```

1 #define NODEID 1  \ \ Numer czujnika
2 #define NETWORKID 1 \ \ Numer sieci
3 #define RECEIVER 0 \ \ Odbiorca (0 = broadcast)
    
```

Istotną cechą zastosowanych bibliotek jest możliwość wykrywania kolizji i automatyczne przejście w stan odbioru w przypadku ich wystąpienia. W praktyce jednak należy zastosować bardziej precyzyjny sposób, by neutralizować występujące kolizje w topologii pozbawionej punktu sterującego.

Transfer danych pomiędzy transceiverem a mikrokontrolerem odbywa się przez interfejs SPI (Rysunek 2.8), zaś mikrokontroler pracuje w trybie

master. Interfejs SPI składa się z dwóch rejestrów przesuwnych połączonych w licznik pierścieniowy i generatora sygnału taktującego, który pochodzi z mikrokontrolera (master). Wejściem danych dla urządzenia master jest linia MISO (zaś wyjściem dla slave).



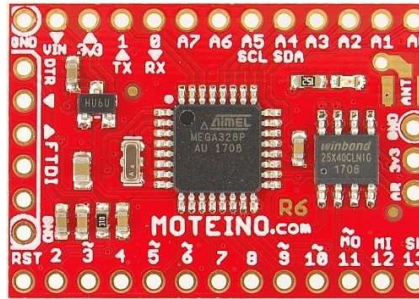
Rysunek 2.8: Interfejs SPI - transmisja danych.

Linia MOSI jest wyjściem dla urządzenia pełniącego rolę master, zaś wejściem dla urządzenia slave. Linia sygnału taktującego SCK jest wejściem dla układu slave, zaś wyjściem dla master. Transmisja danych w interfejsie SPI jest dwukierunkowa, gdyż nadawaniu danych poprzez linię MOSI przez układ pełniący rolę master, zawsze towarzyszy nadawanie danych na linii MISO przez układ slave. W przypadku, gdy jedno z urządzeń nie nadaje, jedna z linii może nie zawierać żadnych informacji.

Urządzenie master może obsługiwać wiele urządzeń slave poprzez interfejs SPI. W tym celu wybiera poszczególne układy podrzędne (slave) za pomocą linii SS w układach podrzędnych ustalając określony stan, jednakże podczas transmisji może być aktywny wyłącznie jeden układ slave, by uniknąć kolizji [3].

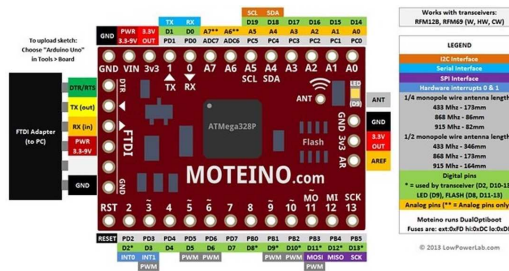
2.5. Moduł pomiarowy (węzeł)

Początkowo próby wykorzystania dostępnych transceiverów wykonano na urządzeniach FS1000A+XY-MK-5V 433 MHz. Pierwsze testy wykazały, że te urządzenia nie nadają się do zastosowania w projekcie ze względu na zbyt dużą odchyłkę częstotliwości, jak również wrażliwość na jakość zasi-



Rysunek 2.10: Widok zmontowanego modułu.

Dodatkowym modulem, nie ujętym na schemacie, jest konwerter USB na TTL, który został podłączony do złącza FTDI modułu mikrokontrolera. Umożliwia on programowanie urządzenia oraz zapewnienie komunikacji z komputerem poprzez złącze USB.



Rysunek 2.11: Moteino - wprowadzenia modułu.

Zakupione płytki PCB wyposażono w niezbędne elementy – w tym moduły radiowe RFM69. Pozwoliło to na zmniejszenie ilości dodatkowych połączeń. Każde z urządzeń pracuje przy napięciu 3.3 V, co pozwoliło wykorzystać akumulator litowo-jonowy wraz z modulem zasilania i ładowania węzła. Początkowo moduły były wyposażone w dodatkowe przetwornice STEP-UP 3.3 V, jednakże pomiary wykazały, że moduł ładowania zapewnia wystarczające napięcie dla mikrokontrolera oraz transceivera. W konsekwencji przetwornice zostały usunięte z projektu.

Oprócz urządzeń akwizycyjnych, do przeprowadzania badań zbudowano



Rysunek 2.12: Pojedynczy moduł pomiarowy.

także urządzenie pomiarowe wyposażone w wyświetlacz i współpracujące z węzłami sieci. Dzięki wyświetlaczowi urządzenie pozwala, bez użycia komputera, w szybki sposób oszacować jakość połączenia z innymi czujnikami, wyświetlając na bieżąco ilość czujników w sieci, odczytane wartości oraz RSSI.



Rysunek 2.13: Zestaw pomiarowy moduły czujników i dodatkowy akwizytor danych.

Podczas pierwszych testów transmisji, badania wykazały, że pomimo zastosowania w dostarczonych bibliotekach detekcji i rozwiązywania kolizji, wciąż występowały pojedyncze oraz wielokrotne kolizje. Konsekwencją tego zachowania było czasowe lub trwale zawieszanie się pracy sieci. Również pojedyncze włączanie modułów, będących w zasięgu, nie zawsze rozpoczynało uruchomienie transmisji. Doprowadzało to do sytuacji nieregularnych zawiesznień sieci w losowych czasookresach, wynoszących od kilkunastu minut do kilkunastu godzin. Problem został rozwiązany przy pomocy algorytmu zaimplementowanego w każdym module, który realizował sygnał „wzbudzenia” sieci i jest generowany na podstawie liczby pseudolosowej, stanowiącej

opóźnienie przed wysłaniem pakietu. Sygnał ten rozwiązał problem, zastępując moduł centralny, który wysyła zapytania do czujników – wzbudzając je do nadawania. Jeśli w sieci aktywny jest wyłącznie jeden moduł i nie odebrał żadnych danych od innych czujników, wysyła on również co pewien czas sygnał aktywacji sieci, dopóki nie zidentyfikuje w sieci innych węzłów. Każdy z modułów generuje na porcie szeregowym dane pomiarowe pochodzące zarówno z wbudowanego czujnika, jak i szereg parametrów pochodzących z innych węzłów.

2.6. Pomiar napięcia akumulatora

Każdy z sensorów wyposażony jest w algorytm pomiaru zasilania, który jest jednym z elementów zwiększenia niezawodności sieci, zaś wymieniane pomiędzy urządzeniami struktury danych, oprócz wartości pobranych z czujników, zawierają także wartość zmierzonego napięcia akumulatora.

Pomiar napięcia wbudowanego w czujnik akumulatora realizowany, jest poprzez odczyt parametru z wejścia analogowego. Zastosowany mikrokontroler (a zatem i jego przetwornik ADC) pracuje przy maksymalnym napięciu 3.3 V, zaś akumulator daje maksymalne napięcie 3.7 V. Należy zatem dostosować napięcie dla ADC za pomocą dzielnika rezystorowego.

Zgodnie z prawem Ohma:

$$V_{out} = V_s \frac{R_2}{R_1 + R_2} \quad (2.5)$$

gdzie:

V_s – napięcie źródłowe,

R_1 – rezystancja pierwszego rezystora, mierzona w omach,

R_2 – rezystancja drugiego rezystora, mierzona w omach,

V_{out} – napięcie wyjściowe mierzone w voltach.

Dla pary rezystorów o rezystancji 1 M, na wyjściu otrzymamy wartość 1.85 V ($2x = 2.7V$) dla w pełni naładowanego akumulatora. Przetwornik ADC w mikrokontrolerze Atmega 328 posiada rozdzielczość 10 bitów, zatem maksymalna próbka pomiaru wynosi 1024, stąd $1024 \cdot 1.85/3.3$ da

w przybliżeniu maksymalną wartość 574 na wejściu pinu analogowego ADC.

Aby przekształcić odczytaną z przetwornika wartość na wolty, należy pomnożyć odczyt przez 0.0032 oraz pomnożyć przez dzielnik – czyli 2. Dodatkowo aby uniknąć błędów pomiaru, należy dokonać jednocześnie kilku pomiarów (doświadczalnie przyjęto 7 pomiarów) oraz wyciągnąć średnią, która jest wynikiem pomiaru.

Listing 2.2: Algorytm pomiaru napięcia akumulatora

```
1 #define AKU_POMIAR(readbat) readbat * 0.00320 * 2
2
3 float battV = 0.0;
4
5 float readBattery ()
6 {
7     unsigned int rbat=0;
8     for (byte i=0; i<7; i++) {
9         rbat += analogRead(A0);
10    }
11    battV = AKU_POMIAR(rbat / 7.0);
12    return battV;
13 }
```

2.7. Dobór anten

Moduły transceiverów nie są fabrycznie wyposażone w anteny. Z tego powodu zasięg modułów ograniczony jest do maksymalnie kilku metrów. W fazie projektowania prototypów okazało się to przydatne, bowiem umożliwiło zbadanie reakcji węzłów na zanik transmisji i dopasowanie parametrów dla algorytmu detekcji innych urządzeń w sieci, bez ich zbytznego oddalania. Jednakże praktyczne wykorzystanie urządzeń, wymagało zastosowania anten przy uwzględnieniu kompromisu, pomiędzy wielkością obudów a zasięgiem, w jakim moduły będą ze sobą współpracować. W urządzeniach przenośnych tego typu, gdzie mamy do czynienia z niewielką mocą nadawczą (enegooszczędność), rodzaj zastosowanej anteny ma fundamentalny wpływ na zasięg – a zatem na prawidłowe funkcjonowanie sieci urządzeń.

Dobór anten nie jest zadaniem trywialnym i wymaga zbudowania i przeprowadzenia pomiarów wielu prototypów. Niniejsza publikacja jest jednak poświęcona zbadaniu możliwości współpracy urządzeń bezprzewodowych, w warunkach pozbawionych punktu centralnego oraz możliwości przekazywania danych modułów między sobą. Mimo wszystko, do celów badaw-

czych transceivery wyposażono w anteny zamontowane wewnątrz obudowy każdego z urządzeń. Zapewniło to wystarczający zasięg do podstawowych pomiarów na potrzeby niniejszego opracowania.

W modułach pomiarowych zastosowano antenę helikalną (Rysunek 2.14) w postaci cewki przylutowanej bezpośrednio do modułu transceivera. Wybór tego typu anten podyktowany był naukową ciekawością sprawdzenia tego typu anten w praktyce. Dużą zaletą tego typu anten są ich niewielkie wymiary i możliwość ich zamocowania wewnątrz niewielkiej obudowy. Wadą zaś - niewielki zasięg urządzeń.



Rysunek 2.14: Antena helikalna zamontowana wewnątrz czujnika.

Tablica 2.4: Dane techniczne zastosowanych anten helikalnych.

Charakterystyka	Wartość
Częstotliwość	433 MHz
VSWR	≤ 1.8
Impedancja wejścia	50 Ohm
Wzmocnienie	2 dBm
Średnica drutu	0.5 mm
Długość	18 mm
Szerokość	4.5 mm

Urządzenie to wyposażono w antenę wykonaną z odcinka przewodu. Jako że każda antena jest przewodnikiem prądu elektrycznego, jej długość musi odpowiadać wielokrotności fali sygnału o danej częstotliwości. Wówczas antena pracuje w rezonansie fali elektrycznej. Długość fali można wyznaczyć dzieląc prędkość fali przez jej częstotliwość:

$$\lambda[m] = \frac{2.99792458 \cdot 10^8 [m/s]}{f[GHz]} \quad (2.6)$$

W praktyce przyjmuje się uproszczony wzór wyrażający częstotliwość w jednostkach MHz, zatem przyjmie on postać:

$$\lambda[m] = \frac{300}{f[Hz]} \quad (2.7)$$

Zatem dla częstotliwości 433 MHz długość fali wyniesie około 69.2 cm. Znając długość fali, można zbudować antenę w postaci przewodu o długości wynikającej z podziału:

$$\text{długość} = \frac{69.2[cm]}{\text{podział}} \quad (2.8)$$

Tego typu szacunkowe wartości przyjęto w niniejszej publikacji, co pozwoliło na zastosowanie anteny w urządzeniu pomiarowym przy użyciu anteny o długości $69.2/4 = 17.3$ [cm] przewodu. Dobór anteny (jej rezonans z falą elektryczną) pozwala na wypromieniowanie sygnału z maksymalną sprawnością, zaś pominięcie dopasowania anteny prowadzi do bardzo niekorzystnych zjawisk, zarówno po stronie toru nadawczego, jak i odbiorczego. Niedopasowanie może prowadzić bowiem do uszkodzenia nadajnika, pogorszeniu czułości toru odbiorczego czy generowania niekorzystnych fal harmonicznnych.

2.8. Oprogramowanie

Oprogramowanie dla mikrokontrolera napisano w języku C++ przy wykorzystaniu bibliotek producenta [16]. Każdy węzeł posiada zdefiniowany adres sieci, klucz szyfrujący dla algorytmu szyfrującego AES oraz identyfikator węzła.

Biblioteka obsługi RFM69 wymaga zdefiniowania numeru sieci (0-256) oraz adresu urządzenia (węzła) z zakresu 1-256. Adres o numerze 0 to specjalny adres rozgłoszeniowy, używany do wysyłania danych do wszystkich węzłów w sieci. Zatem adresując pakiet w określonej sieci na adres 1, wszystkie pozostałe węzły w sieci odbiorą ten pakiet.

Listing 2.3: Parametry konfiguracyjne modułu pomiarowego

```

1 #define NETWORKID 0
2 #define MYNODEID 1
3 #define TONODEID 0
4 #define FREQUENCY RF69_433MHZ
5 #define ENCRYPT true
6 #define ENCRYPTKEY "16bitowykluczszyfrujacy"

```

Domyślnie włączone jest też szyfrowanie AES oraz zdefiniowany klucz szyfrujący – identyczny w każdym firmware urządzenia. Dzięki temu przesyłane dane są chronione przed nieautoryzowanym dostępem oraz podsłuchem.

Następnie zostały zdefiniowane urządzenia podłączone do mikrokontrolera. Każdy z modułów przekazujących dane do innych modułów wyposażony jest w 3 diody LED podłączone do wyjść cyfrowych: nadawanie, odbiór oraz dioda informująca o włączeniu zasilania. Dioda ta sygnalizuje także ewentualne błędy odczytu, np. błędnego odczytania struktury danych lub odebraniu struktury o nieprawidłowym rozmiarze oraz wejścia sygnału analogowego do pomiaru napięcia akumulatora. Ze względu na dzielnik napięcia, odczyt podlega dodatkowemu przeliczeniu. Pomiar odbywa się na wejściu analogowym A0 mikrokontrolera podłączonym do akumulatora poprzez dzielnik rezystorowy [13].

Listing 2.4: Stałe i zmienne fizycznych urządzeń oraz parametry algorytmów

```

1 #define TXLED 5
2 #define RXLED 3
3 #define PWLED 4
4 #define AKU_POMIAR(readbat) readbat * 0.00320 * 2
5 float battV = 0.0;

```

W programie mikrokontrolera zdefiniowano także stałą MAXTIMEOUT, która steruje algorytmem wygaszania nieistniejących czujników w sieci. Im większa wartość, tym moduły dłużej przechowują w swojej strukturze czujniki, które przestały działać. Wartość ta została dobrana eksperymentalnie i można ją modyfikować w zależności od zastosowania modułów. Oprogramowanie można także przełączyć w tryb promiscuousMode, co oznacza, że moduły przełączają się w tryb nasłuchu odbierają wszystkie pakiety z danej sieci.

Struktura pojedynczego rekordu listy (przesyłany w jednym komunikacie) jest następująca:

Listing 2.5: Struktura danych pojedynczego węzła

```
1 struct device
2 {
3     uint8_t ident;
4     float value;
5     float batt;
6     unsigned int timeout;
7     struct device *next;
8 };
```

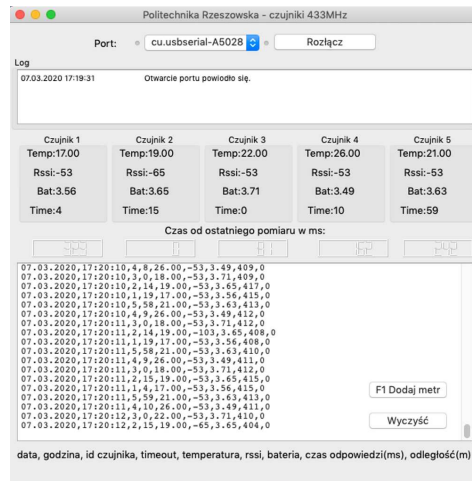
Zmienna **ident** jest numerem węzła, w związku z tym identyfikuje urządzenie w sieci. Dzięki temu, każde z urządzeń, które odbierze komunikat, jest w stanie zidentyfikować jego źródło. **Value** przechowuje wartość odczytaną z czujnika temperatury. Dla uproszczenia wartość ta odczytywana jest z obecnego czujnika w module RFM69, przez co stanowi jedynie wartość badawczą, bowiem pomiar może być zawyżony, ze względu na nagrzewanie się modułu wewnątrz obudowy. Urządzenia można wyposażać w zewnętrzne czujniki mierzące różne parametry – w zależności od potrzeb. **Timeout** jest nieustaloną jednostką czasu, jaki upłynął od czasu otrzymania komunikatu od urządzenia. Liczba ta jest stale zwiększana w pamięci modułu, w zależności od ilości pętli odbioru do czasu osiągnięcia maksymalnej wartości lub zamieniana/zerowana w przypadku nadejścia nowych danych. Jeśli urządzenie otrzyma dane o niższej wartości timeout, uaktualnia dane o danym identyfikatorze w strukturze czujników. Każdy moduł wysyła dane do innych modułów wysyłając pakiety ze strukturami zawierającymi inkrementowane wartości timeout. Dołącza jednocześnie własny rekord z wartością 0, co oznacza, że rekord pochodzi ze źródła o danym identyfikatorze.

2.9. Oprogramowanie pomiarowe i moduł pomiaru

Złożeniem projektu było zbudowanie urządzeń, które pozwalają na bezpośrednio pobranie parametrów z każdego pojedynczego węzła poprzez złącze USB. Każdy z modułów generuje dane w uniwersalnym formacie JSON, zatem w prosty sposób dane mogą być wizualizowane na stronie www. Zestaw zawiera także odbiornik, który generuje na porcie szeregowym dane w formacie CSV o strukturze: id, wartość, timeout, RSSI oraz odległość (fizyczna odległość oddalenia od czujnika będącego celem akwizycji). Urządzenie posiada dwa przyciski. Wciskanie pierwszego, w miarę oddalania się od źródła pomiaru inkrementuje licznik odległości, co znacznie ułatwiło uzyskanie wyników pomiaru. Dzięki temu, wygenerowany CSV można wizualizować w postaci wykresu. Urządzenie generuje pojedynczy CSV, uwzględniając wszystkie czujniki w sieci, zatem utworzenie pliku z pomiarów, pozwala na

porównanie wyników pomiarów zasięgu oraz ich analizę w nietypowych rozkładach pomieszczeń. W badaniu, jako jednostkę odległości przyjęto metr. Dodatkowy odbiornik zbudowany jest podobnie jak każdy czujnik, z tą różnicą, że nie wysyła danych do sieci i jest wyposażony w wyświetlacz LCD 2x16 znaków.

Dla przeprowadzenia testów przygotowano także program do wyświetlania pobranych parametrów (Rysunek 2.15). Dzięki temu pomiarów dokonywać można zarówno podłączając laptopa do portu USB dowolnego czujnika, jak i wyświetlając dane zdalnie, za pomocą dodatkowego urządzenia.



Rysunek 2.15: Oprogramowanie komputerowe do akwizycji sieci czujników.

Oprogramowanie współpracujące z portem USB można podłączyć do dowolnego czujnika w sieci. Współpracuje ono z czujnikami poprzez port USB i dekoduje format JSON. Oprogramowanie zostało utworzone przy pomocy języka C++ [17] oraz biblioteki Qt [7]. Dzięki temu może zostać skompilowane na dowolnym systemie operacyjnych, gdzie dostępna jest biblioteka Qt. Oprogramowanie do akwizycji tworzy także log w formacie CSV, który można zapisać do pliku i analizować za pomocą dowolnego oprogramowania zewnętrznego. W programie zaimplementowano także liczniki, które zliczają czas (ms) od ostatniej akwizycji każdego z czujników (niższa wartość oznacza aktualniejsze dane). Przekroczenie wartości krytycznej czasu powoduje, że licznik wskaże wartość -1. Dzięki temu, podczas analizowania danych, wnioskować można o utracie zasięgu lub uszkodzeniu mo-

dułu. Przy wyłączeniu lub uszkodzeniu któregoś z modułów w sieci, wzrasta timeout w innych modułach w sieci, które przechowują strukturę uszkodzonego czujnika, a następnie wartość licznika, dzięki temu można zbudować algorytm informujący o możliwości wystąpienia problemu z konkretnym czujnikiem, lecz jednocześnie dane uszkodzonego czujnika będą dostępne w pamięci innych czujników, co daje realną szansę odczytu ostatnich parametrów przez potencjalnym uszkodzeniem.

2.10. Pomiary

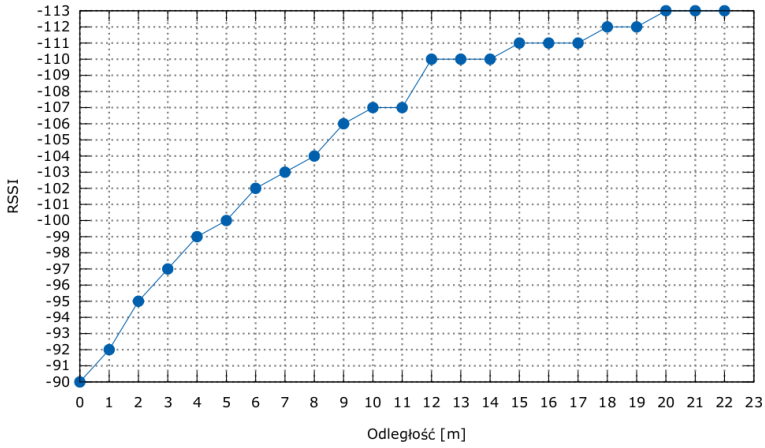
Testy systemu czujników przeprowadzono w wielokondygnacyjnym budynku oraz na otwartym terenie, mierząc jedynie zasięg pojedynczego urządzenia ze zbudowanym specjalnie do tego celu odbiornikiem, który generuje dane w formacie CSV.

Sieci tego typu podlegają takim pomiarom, jak efektywność (rozumiana jako procent pakietów odebranych w stosunku do pakietów nadanych), dynamika czy charakterystyka przepustowości w funkcji odległości. Jednakże na potrzeby tej części badań zrezygnowano z tych pomiarów, analizując jedynie zasięg urządzeń. Zastosowane w projekcie czujniki temperatury nie były przedmiotem badań, stąd wyniki pomiarów temperatury także nie stanowiły przedmiotu badawczego. Celem eksperymentu naukowego na tym etapie było pozyskanie wiedzy, na temat działania sieci bez modułu zarządzającego oraz wyznaczenie potencjalnego zasięgu dla dalszych badań.

Pierwsze testy odbyły się w dwukondygnacyjnym budynku, w którym znajdują się biura, wyposażone w stanowiska komputerowe. Nadajnik został umieszczony w serwerowni, w której występują duże zakłócenia od 4 fizycznych serwerów oraz urządzeń aktywnych sieci. Dodatkowo korytarzami przebiegają instalacje elektryczne i logiczne, zatem warunki pomiaru uznać można za przemysłowe. Badanym parametrem podczas testów był zasięg czujników do czasu pojawienia się pierwszych zakłóceń.

Test 1 (Rysunek 2.16) polegał na zmierzeniu zasięgu pomiędzy czujnikami przy wykorzystaniu anten wbudowanych w urządzenia (antena helikalna przylutowana do płytki modułu i znajdująca się wewnątrz obudowy każdego z czujników).

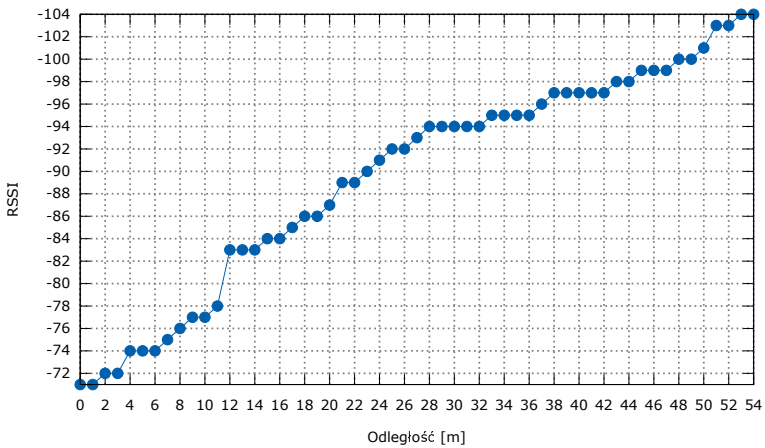
Pomiary wykazały maksymalny zasięg urządzeń na poziomie 20 me-



Rysunek 2.16: Wykres pomiaru zasięgu pomiędzy czujnikami.

trów na terenie otwartym. Podobne testy wykazały o połowę mniejszy zasięg w budynku, gdy czujniki znajdowały się w pomieszczeniu z betonową ścianą.

Test 2 (Rysunek 2.17) polegał na przeprowadzeniu podobnego doświadczenia, jednak realizowanego pomiędzy pojedynczym czujnikiem, wyposażonym w antenę cewkową a modulem akwizycji danych, wyposażonym w antenę, w postaci przewodu zamontowanego wewnątrz obudowy.



Rysunek 2.17: Wykres pomiaru zasięgu pomiędzy czujnikiem a modulem akwizyjnym.

Doświadczenie, przy wykorzystaniu jako jednej z anten przewodu o dobranej długości, wykazało ponad dwukrotnie lepszy zasięg wynoszący 54 m.

2.11. Podsumowanie

Zagadnienia opisane w niniejszej publikacji dotyczą sposobu doboru i optymalizacji charakterystyk elementów stosowanych do budowy bezprzewodowych sieci czujników. Projektowanie oprogramowania dla sieci bezprzewodowych pozbawionych punktu centralnego nie jest zadaniem trywialnym i wymaga przeprowadzenia wielu eksperymentów. Mimo wszystko etap budowy i uruchomienia kilku węzłów został zakończony. Podstawowym wnioskiem przy budowie kolejnych prototypów powinno być zapewnienie zewnętrznych anten, co umożliwi zwielokrotnienie zasięgu. Zastosowane anteny, wykonane z drutu Cu w postaci cewki narażone są na mechaniczne odkształcenia ze strony innych elementów znajdujących się wewnątrz obudowy urządzeń prototypowych. System sensorów był testowany przez kilka dni w warunkach rozproszenia czujników w odległości 10-30 metrów, przy czym można założyć, że krytycznym dla działania sieci nie jest zasięg, który przy odpowiednio dobranych antenach wynieść może nawet kilkaset metrów w warunkach otwartej przestrzeni, lecz ograniczenia wynikające z budowy transceiverów RFM69. Testy wykazały poprawne działanie sieci nawet w przypadku, gdy urządzenia pomiarowe były pojedynczo lub zbiorowo wyłączane podczas procedury testowej. Zastosowany algorytm identyfikacji bezczynności/awarii pozwolił na odczyt danych również z wyłączonych/uszkodzonych czujników przez określony czas, dzięki mechanizmowi buforowania we wszystkich węzłach danych z innych czujników przez ustalony czas. Po ponownym włączeniu czujników następuje automatyczne wznowienie akwizycji danych. Na podstawie pobranych parametrów, możliwe jest również zaprogramowanie mechanizmu wnioskowania o utracie jednego lub n -czujników. Warto także podkreślić, że rozwiązanie polegające na zastosowaniu dynamicznie alokowanych tablic, przechowujących dane ze wszystkich urządzeń, stanowiąc wartościowe doświadczenie naukowe.

W opinii autora, należy prowadzić dalsze badania i opracować algorytm doskonalszego kolejgowania danych oraz doskonalić dynamiczne dostosowanie się sieci do priorytetyzacji nadawania. Dzięki temu, na podstawie parametrów, czujniki wymieniałyby między sobą dodatkowe informacje konfiguracyjne, co pozwoliłoby na lepszą adaptację urządzeń, w zależności od liczby czujników w sieci. W opinii autora może mieć to niebagatelny wpływ na

optymalne wykorzystanie pasma transmisji oraz oszczędność energii. Dobrym kierunkiem, podczas dalszych badań nad zastosowaną siecią, może być wdrożenie doskonalszych algorytmów reorganizacji sieci w przypadku wyeliminowania jednego lub n czujników w sieci bądź czasowego wyłączenia nieużywanych węzłów wtedy, gdy działają nieprawidłowo lub znajdują się poza zasięgiem [20].

Ze względu na ograniczenia, badania wykonano jedynie na pięciu czujnikach, co nie pozwoliło na zbadanie w praktyce możliwej maksymalnej ilości czujników pracujących bez modułu zarządzającego. Dodatkowo należy podjąć dalsze badania i wyposażać czujniki w bardziej złożone algorytmy rozpoznawania innych urządzeń będących w zasięgu. Następnie na tej podstawie, zbudować bardziej wydajny algorytm generowania opóźnień transmisji, przy zastosowaniu bardziej wydajnych i posiadających więcej zasobów sprzętowych mikrokontrolerów i zaprogramowaniu mechanizmu routingu. Istotnym ograniczeniem systemu jest występujący problem zasilania. W sieci bez kontrolera głównego, czujniki pracują bowiem bez trybu uśpienia, gdyż muszą nasłuchiwać innych węzłów w sieci oraz analizować bezczynność. Po przekroczeniu tego czasu, urządzenia muszą nadać komunikat inicjujący potencjalnie istniejące, sąsiednie czujniki do nadawania. Nie jest to zatem rozwiązanie energooszczędne, choć niewielki pobór prądu pozwala na wdrożenie zasilania z ekologicznych źródeł energii.

Badania wykazują duży potencjał sieci bezprzewodowej w warunkach przemysłowych oraz poza przemysłowym (domowych), gdzie dynamika systemu nie jest wartością krytyczną – w chwili obecnej jest ona bowiem wystarczająca dla zastosowań nawet w określonych obszarach krytycznych. Wbudowana kontrola błędów oraz szyfrowanie pozwala na bezpieczną akwizycję danych w miejscach, gdzie niewskazane lub niemożliwe jest prowadzenie przewodów. Zbudowana sieć bezprzewodowych czujników, może znaleźć zastosowanie w akwizycji danych, pochodzących z systemów cyfrowego domu ale także w tworzeniu redundantnych zestawów pomiarowych w obszarze przemysłu, monitoringu i autoryzacji dostępu, czy zdalnej akwizycji danych w zabezpieczonych pomieszczeniach. Dzięki autonomiczności węzłów, po zastosowaniu czujników innych wartości fizycznych, moduły mogą zostać rozmieszczone w dowolnej konfiguracji – w tym także obok siebie, dając wymagany poziom redundancji dla systemu pomiarowego.

Bibliografia

- [1] R. Anderson, D. Cervo. *Arduino dla zaawansowanych*. Helion, Gliwice, 2013.
- [2] T. Dicola. *AdafruitRFM69 Library Documentation*. Adafruit, New York, 2020.
- [3] T. Francuz. *AVR. Układy peryferyjne*. Helion, Gliwice, 2014.
- [4] F. Gao, H. Wen, L. Zhao, Y. Chen. Design and optimization of a cross-layer routing protocol for multi-hop wireless sensor networks. *International Conference on Sensor Network Security Technology and Privacy Communication System*, strony 5–8, 2013.
- [5] B. Harpence. *Routing i switching. Praktyczny przewodnik*. Helion, 2011.
- [6] K. Hyung-Sin, H. Jin-Seok, L. Yong-Hwan. Scalable network joining mechanism in wireless sensor networks, 2012.
- [7] G. Lazar, R. Penea. *Mastering Qt5. Second Edition*. Helion, Gliwice, 2018.
- [8] Y. Lei, Y. Zhang, Y. Zhao. The Research of Coverage Problems in Wireless Sensor Network. *Automatyka*, 13(3), 2009.
- [9] L. Li, Y. Zhu, Y. Yu. Link scheduling and Data Forwarding in Wireless Sensor Networks of long chains tree topology. *IET International Conference on Wireless Sensor Network 2010*, strony 297–300, 2010.
- [10] Y. Liu, X. Jiang. A Protocol Model for Wireless Sensor Network. *International Conference on Networks Security, Wireless Communications and Trusted Computing*, strony 588–591, 2009.
- [11] Y. Liu, K. Tong, X. Qiu, Y. Liu, X. Ding. Wireless Mesh Networks in IoT networks. *International Workshop on Electromagnetics: Applications and Student Innovation Competition*, 2017.
- [12] LowPowerLab. *Computing higher-order moments online*, 2017.
- [13] S. Monk. *Elektronika z wykorzystaniem Arduino i Raspberry PI*. Helion, Gliwice, 2017.

- [14] W. Nowicki. *Rozproszone systemy pomiarowe*. WKŁ, Warszawa, 2006.
- [15] R. Olejnik. Przegląd rozwiązań sprzętowo-programowych pozwalających na budowę bezprzewodowych sieci o topologii kratowej. *Metody Informatyki Stosowanej*, 1/2009(18), 2009.
- [16] S. Prata. *Język C. Szkoła programowania*. Helion, Gliwice, 2016.
- [17] S. Prata. *Język C++. Szkoła programowania*. Helion, Gliwice, 2019.
- [18] D. Sankowski, W. Mosorow. Zastosowanie Wireless Sensor Network Simulator do badań sieci sensorowych. *Automatyka*, 13(3), 2009.
- [19] C. Wtorek. *Bezprzewodowe sieci sensorowe*, 2020.
- [20] M.S. Zahraie, A.Z. Farkhady, A.T. Haghghat. 2009 11th International Conference on Computer Modelling and Simulation. *Increasing Network Lifetime by Optimum Placement of Sensors in Wireless Sensor Networks*, strony 611–616, 2009.

Building a decentralized network of wireless sensors to acquire measurement data from production lines

Abstract: Wireless Sensor Networks can be used in the field of industry and digital home for the acquisition of data from many measuring devices. Typical sensor networks are built in a star topology, with a central management point, which is also a data acquisition point. Presented publication provides a description of wireless network sensors construction with the use of transceivers operating at 433MHz. Such constructed network is resistant to failures, thanks to the elimination of the central point, while each node can be used at the same time as an intermediate node and thanks to the algorithm of a dynamic array with data from other nodes as an acquisition point. What is more, the network allows for the analysis of failures in one or more nodes in the network.

3. Budowa bezprzewodowej sieci czujników w oparciu o moduły Wi-Fi

ADAM MICHAJLYSZYN¹

POLITECHNIKA RZESZOWSKA, A.MICHAJLYSZ@PRZ.EDU.PL

Streszczenie Aktualnie dostrzegamy trwały trend na świecie, dotyczący stałego wzrostu liczby urządzeń z komunikacją bezprzewodową. Dzięki temu następuje szybki i wszechstronny rozwój wykorzystywania Internetu rzeczy (Internet of Things). Bezprzewodowa sieć czujników WSN (Wireless Sensor Network) dotyczy grupy rozproszonych czujników, połączonych z tą samą siecią bezprzewodową. W tej grupie, zdobywającym coraz większą popularność stał się moduł ESP8266, wyprodukowany przez chińską firmę Espressif. Z uwagi na swoje możliwości oraz niewielkie rozmiary, a zarazem niskie zużycie energii znalazł zastosowanie w wielu dziedzinach życia. Warto wspomnieć chociażby o nawadnianiu, rolnictwie, monitorowaniu zjawisk atmosferycznych a przede wszystkim zarządzaniu procesami produkcyjnymi i technologicznymi. Dane środowiskowe są wizualizowane w postaci graficznej (wykresy diagramy) i pozwalają użytkownikowi na kontrolowanie i zdalne zarządzanie, nawet przy użyciu smartfonu. Korzystając z urządzeń monitorujących i sterujących za pomocą istniejącej infrastruktury Wi-Fi można znacznie zmniejszyć koszty produkcji przy niezmiennym poziomie wydajności, co pozytywnie wpływa na wiele dziedzin gospodarki, a także ochronę środowiska. W artykule przedstawiono krótką charakterystykę płytki z modułem ESP8266 (Node-Mecu) oraz jej praktyczne zastosowanie do monitorowania tempera-

¹ORCID: 0000-0002-6458-387X, Wydział Mechaniczno-Technologiczny Politechniki Rzeszowskiej, Kwiatkowskiego 4, 37-450 Stalowa Wola

tury z przesyłaniem wyników na platformę ThingSpeak.com gdzie otrzymane dane są częściowo przetwarzane.

3.1. Wprowadzenie

Aktualnie obserwujemy na świecie stały wzrost liczby urządzeń z komunikacją bezprzewodową. Zauważamy ich dominację w kolejnych obszarach życia, wynikającą z postępującej redukcji kosztów produkcji oraz minimalizacji gabarytów, zarówno sprzętu komputerowego, jak i urządzeń komunikacyjnych. Ten trend powoduje szybki i wszechstronny wzrost wykorzystywania Internetu rzeczy (IoT). Bezprzewodowa sieć czujników WSN (Wireless Sensor Network) określa sieć zbudowaną z rozproszonych czujników połączonych jedną siecią bezprzewodową. WSN zbudowana jest z kilku węzłów i bramy lub węzła centralnego, połączonych między sobą bezprzewodowo. Każdy węzeł składa się z urządzenia nadawczo-odbiorczego i anteny.

Tanim przedstawicielem zdobywającym coraz większą popularność stał się moduł ESP8266, wyprodukowany przez chińską firmę Espressif. Z uwagi na swoje możliwości, niewielkie rozmiary, a zarazem niskie zużycie energii znalazł zastosowanie w wielu dziedzinach życia. Wykorzystywany jest m.in. w systemach nawadniających terenów zielonych. Temat szeroko opisał Suit Thakare i wsp. [17], którzy w swoim artykule przedstawili inteligentny system nawadniania, zapewniający automatyzację w gospodarstwie. Zastosowany system czujników z płytą Arduino i modulem Wi-Fi, dostarcza informacji o parametrach gleby. Na tej podstawie następuje uruchomienie systemu nawadniania. Arduino zbiera dane ze wszystkich czujników i łączy je z chmurą. Główną zaletą systemu jest to, że właściciel może zdalnie i na bieżąco monitorować farmę.

Moduł ESP8266 wykorzystywany jest również w zarządzaniu szklarniami [7]. Ziyu Wan i wsp. [18] przedstawili zastosowanie modułu na płycie NodeMecu, jako jednostce centralnej kontrolującej parametry środowiskowe, temperaturę powietrza, stężenie CO₂ i wilgotność gleby. Przy użyciu protokołu HTTP, ESP8266 NodeMecu, podłączony do WI-FI komunikuje się z platformami IoT - Thingspeak i Bylnk. Dane środowiskowe są wizualizowane w postaci szeregów i pozwalają użytkownikowi na kontrolowanie i zdalne zarządzanie szklarniami, nawet przy użyciu smartfonu. Moduł znalazł zastosowanie także w stacjach pogodowych, w monitorowaniu temperatury, wilgotności względnej, ciśnienia, opadów, prędkości wiatru itp. [13],

[16], [4]. Moduł ten wykorzystywany jest również w kontrolowaniu zanieczyszczeń powietrza [8], co w dzisiejszych czasach ma szczególnie istotne znaczenie. Szeroko wykorzystuje się go w przemyśle do monitorowania procesów produkcyjnych. Komal i Prachi [15] w swoim artykule z Międzynarodowej Konferencji na temat I-SMAC 2017 opisują, jak za pośrednictwem połączenia internetowego można prowadzić nadzór nad procesami przemysłowymi. W zaproponowanym systemie wykorzystują telefon komórkowy z Androidem lub laptop do monitorowania liczników energii elektrycznej, prędkości obrotowej silników prądu stałego oraz w oparciu o sterowniki PLC monitorowania mieszania farb i kontroli poziomu cieczy. Natomiast Feng Zhang i wsp. [20] w swojej pracy przedstawiają system monitorujący, wykorzystywany podczas ciągłego odlewania stali. Zastosowali IoT w celu gromadzenia danych sensorycznych, takich jak temperatura i ciśnienie, które przekazywane są do centrum monitorowania. Takie dane stanowią poważne wyzwanie do monitorowania zmiennych w czasie rzeczywistym i ułatwiają podejmowanie decyzji, w szczególności wtedy, gdy czas ma istotne znaczenie. Zaproponowany system monitorowania składa się z czterech warstw:

1. wykrywania,
2. sieci,
3. zasobu usługowego,
4. aplikacji.

W systemie integrują się różne techniki przetwarzania danych, w tym konfiguracja protokołu, wersja i filtrowanie danych wraz z ich konwersją. System wdrożono i zademonstrowano przy rzeczywistej linii produkcyjnej do ciągłego odlewania stali. Zintegrowany został z platformą TeamCenter. Otrzymane wyniki wskazały, że zaproponowane rozwiązanie dobrze radziło sobie w rzeczywistym środowisku przemysłowym. Ravi Kishore Kodali i Aditya Valdas [6] w swoim artykule przedstawili przy użyciu mikrokontrolera NodeMecu z włączonym chipem Wi-Fi ESP8266, system monitorowania w zakładzie produkcyjnym. Punktem ich zainteresowań stał się człowiek oraz jego bezpieczeństwo w miejscu pracy. Używając kilku czujników: czujnika temperatury i wilgotności (DHT), czujnika ultradźwiękowego (HC-04) i czujnika dymu (MQ2) stale monitorowali miejsca pracy i przesyłali dane do platformy IoT. Wykorzystali Losant, jedną z największych chmur obsługującą dane, a przy tym gwarantującą ich bezpieczeństwo, jak również wizualizację i przetwarzanie zgodnie z oczekiwaniami klienta. Dane

do chmury przesyłano przy pomocy protokołu MQTT (Message Queue Telemetry Transport).

Internet rzeczy, staje się również coraz bardziej istotny w naszym codziennym życiu. Przede wszystkim stosuje się go do kontroli, monitorowania i zarządzania technologią, z której korzystamy na co dzień [9], [5], [12], [10], [2], [14], [1]. Oznacza to, że urządzenia te projektowane są w taki sposób, aby były proste w instalacji oraz intuicyjne w zarządzaniu przez odbiorcę. Zastosowanie tego typu urządzeń ma na celu oszczędność energii i zasobów, z którymi mamy do czynienia na co dzień. W dzisiejszym świecie korzystając z urządzeń monitorujących i sterujących za pomocą istniejącej infrastruktury Wi-Fi można znacznie zmniejszyć koszty produkcji przy niezmiennym poziomie wydajności, co pozytywnie wpływa na wiele dziedzin gospodarki a także ochronę środowiska.

3.2. Charakterystyka modułu bezprzewodowego Wi-Fi na przykładzie ESP8266

Moduł ESP8266 bazuje na 32-bitowym procesorze RISC Tensilica L106. Standardowo taktowany jest 80 MHz zegarem (występuje również wersja o taktowaniu 160 MHz) i składa się z cyfrowych interfejsów peryferyjnych, przełączników antenowych, wzmacniacza mocy, wzmacniacza odbioru o niskim poziomie szumów, filtrów i modułu zarządzania energią. Posiada 17 wejść/wyjść cyfrowych (GPIO 0-16), jednak można użyć tylko 11, ponieważ 6 z nich (GPIO 6-11) służy do podłączenia układu pamięci flash. Próba użycia jednego z tych wejść może zakończyć się zawieszeniem programu. Wyróżnić możemy następujące złącza:

- GPIO 1 i 3 – TX i RX złącza portu szeregowego (UART0)
- GPIO 2 i 8 – TX1 i RX1 złącza portu szeregowego (UART1)
- GPIO 13 i 15 – RTS0 i CTS0 pełnią funkcje sprzętowej kontroli przepływu, alternatywnie można je używać jako TX0 i RX0
- GPIO 4 i 5 – pełnią funkcję SDA (Serial Data Line) i SCL (Serial Clock Line)
- GPIO 14 – CLK
- GPIO 12 – MISO

- GPIO 13 – MOSI
- GPIO 15 – SS (Slave Select)
- GPIO 0 – napięcie 3,3V
- GPIO 16 – złącze służące do wybudzania z trybu głębokiego uśpienia.

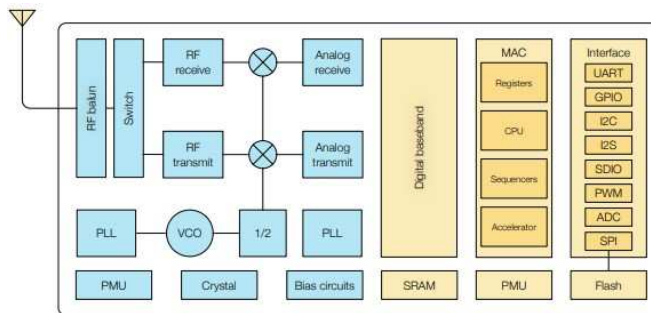
Wszystkie wymienione elementy znajdują się w jednym module o niewielkich gabarytach, w niektórych wersjach nawet o powierzchni ok. 2-3 cm². Dzięki wysoce zintegrowanym funkcjom na chipie i minimalnej liczbie elementów zapewnia niezawodność, kompaktowość oraz wytrzymałość. Natomiast odporność na szeroki zakres temperatur pozwala mu skutecznie funkcjonować w zróżnicowanych środowiskach. Zastosowany procesor wymaga wyjątkowo niskiego zapotrzebowania na energię a moduł w stanie spoczynku może pobierać jedynie około 10 μ A. Zasilany jest standardowo 3.3 V. Należy zatem zwrócić uwagę podczas podłączania zasilania, aby napięcie na wejściu mieściło się w granicach jak podaje producent od 3.0 V~3.6 V, gdyż większe napięcie może spowodować nieodwracalne uszkodzenie. System operacyjny (RTOS) i układ Wi-Fi umożliwiają przekierowanie 80% mocy na obsługę programowania i aplikacji zainstalowanych przez użytkownika, co potwierdza ich wysoką energooszczędność. Moduł obsługuje standard Wi-Fi 802.11 b/g/n, ale pracuje jedynie na częstotliwości 2.4 GHz. Istotnym aspektem jest fakt, że urządzenie może pracować zarówno jako klient sieciowy, jak i punkt dostępowy (access point) w tym samym czasie. Posiada zintegrowany stos protokołów TCP/IP. Zaprojektowany został z myślą o zastosowaniu w mobilnych urządzeniach, elektronice i aplikacjach IoT. Niskie zużycie energii osiąga dzięki zastosowaniu kilku technologii. Architektura powodująca oszczędności wynika z kilku trybów pracy:

1. *Tryb aktywny* – chip radia pozostaje włączony, chip może odbierać, przesyłać lub nasłuchiwać.
2. *Tryb uśpienia modemu* – CPU działa, Wi-Fi i radio są wyłączone. Tryb uśpienia modemu jest używany w aplikacjach wymagających pracy procesora, jak w PWM (Pulse Width Modulation) lub Aplikacjach I2S. Zgodnie ze standardami 802.11 (U-APSD – Unscheduled Automatic Power Save Delivery), wyłącza modem Wi-Fi przy zachowaniu połączenia Wi-Fi bez transmisji danych w celu optymalizacji mocy konsumpcji. Np. W DTIM3, utrzymywanie snu 300 ms z budzeniem trwającym 3 ms do odbioru Pakietu AP Beacon wymagają w odstępach około 15 mA prądu.

3. *Tryb uśpienia* – procesor i wszystkie urządzenia peryferyjne są wstrzymane. Wszystkie wzbudzenia (MAC, host, zegar RTC lub zewnętrzne przerwania) wyzwała chip. W trybie lekkiego uśpienia procesor może być zawieszony w aplikacjach, takich jak przełącznik Wi-Fi. Bez transmisji danych, obwód modemu Wi-Fi można wyłączyć, a procesor zawiesić, aby oszczędzać energię zużycia. Zgodnie ze standardami 802.11 (U-APSD). Np. W DTIM3, utrzymywanie snu 300 ms z wybudzeniem 3 ms w celu odebrania pakietów Beacon AP w interwale wymaga około 0,9 mA prądu.
4. *Tryb głębokiego uśpienia* – działa tylko zegar czasu rzeczywistego i wszystkie inne części chipa są wyłączone. W trybie głębokiego uśpienia Wi-Fi jest wyłączone. Do aplikacji z dużymi opóźnieniami między danymi transmisji, np. czujnik temperatury wykrywający temperaturę co 100 s, śpi przez 300 s i budzi się, aby połączyć się z punktem dostępowym (trwa około 0,3 ~ 1s), ogólny średni prąd jest mniejszy niż 1mA. Prąd 20 μA jest pobierany przy napięciu 2,5 V [1].

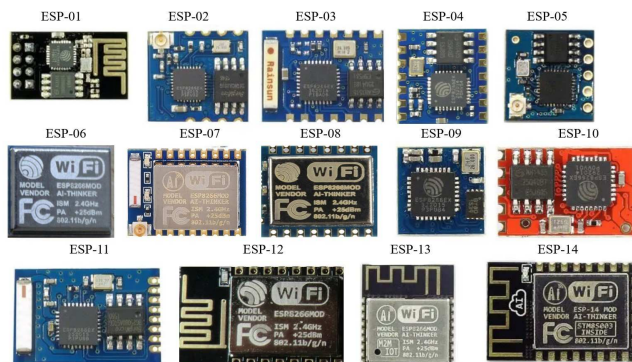
Zaprezentowane tryby pracy pozwalają na zwiększenie czasu pracy na zasilaniu bateryjnym.

Schemat blokowy modułu ESP8266 przedstawia Rysunek 3.1.



Rysunek 3.1: Schemat blokowy ESP8266 [3].

Wbudowany interfejs UART daje możliwość programowania modułu Arduino lub innego mikrokontrolera. Moduły Wi-Fi z układem ESP8266 są oferowane w różnych wariantach wykonania. Główne różnice polegają na odmiennej liczbie dostępnych wyprowadzeń I/O, rodzaju zintegrowanej anteny Wi-Fi lub pamięci Flash. Przedstawiono je na Rysunku 3.2 oraz Tabeli 3.1.



Rysunek 3.2: Wersje modułu ESP8266.

Tablica 3.1: Zestawienie podstawowych wersji modułu ESP8266.

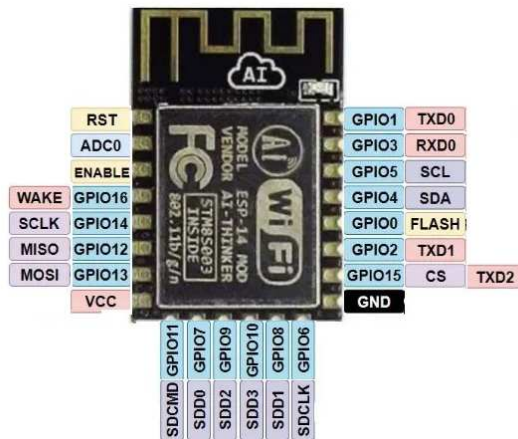
Nazwa układu	Pamięć Flash	Antena/gniazda	Liczba GPIO
ESP-01	512 kB	PCB	3
ESP-02	512 kB	IPX RF	3
ESP-03	512 kB	Ceramiczna	7
ESP-04	512 kB	IPX RF	6
ESP-05	512 kB	IPX RF	0
ESP-06	1024 kB	brak	7
ESP-07	4096 kB	IPX RF/ceramiczna	9
ESP-08	1024 kB	brak	7
ESP-09	1024 kB	brak	6
ESP-10	512 kB	brak	0
ESP-11	512 kB	Ceramiczna	2
ESP-12	4096 kB	PCB	9+ADC
ESP-13	16384 kB	PCB	9+ADC
ESP-14	8 kB	PCB	10+ADC

Istnieje wiele sposobów programowania ESP8266. Producent dostarcza w tym celu oficjalne SDK. Moduł może być programowany w języku C++, Lun, a nawet MicroPython. ESP można potraktować również, jak zwykły moduł komunikacyjny, z którym wymiana informacji następuje przy użyciu komend AT. Jeden z prostszych sposobów stanowi programowanie przy

użyciu Arduino IDE.

Z uwagi na niską cenę, a zarazem duże możliwości układ ten wzbudza zainteresowanie wielu programistów i społeczności open source. Oprócz samych płytek producent zaleca i wskazuje środowiska rozwojowe takie jak Mongoose.os, MicroPython, NodeMcu, Arduino, Platformio.org i Zerynth.

Aktualnie najnowszą płytką jest moduł ESP8266 14E. Moduł ten ma łącznie 22 piny. Opis pinów przedstawia Rysunek 3.3 oraz Tabela 3.2 .



Rysunek 3.3: Opis podłączeń poszczególnych wyjść z modułu ESP8266 14E.

Tablica 3.2: Opis poszczególnych wyjść z modułu ESP8266 14E.

Lp.	Liczba wyjść	Etykieta wyjścia	Opis poszczególnych wyjść
1.	17	GPIO	Zakres styków GPIO od GPIO0do GPIO16 obejmuje piny interfejsu SPI, 12C, SDIO, UART
2.	1	ADC	10-bitowy przetwornik analogowo-cyfrowy
3.	1	VCC	Napięcie zasilania 3,3 V
4.	1	GND	Pin uziemiający
5.	1	RST	Reszta PIN
6.	1	ENABLE	Chip Enable Pin

3.3. Zasięg i jakość sygnału w zależności od zastosowanej anteny

Yoppy i wsp. [19] porównali RSSI (Received Signal Strength Indicator) modułów ESP8266. Badaniu poddali cztery moduły z różnego rodzaju antenami, oznaczając je kolejno literami A-D.

A – moduł z anteną PCB (Antena z obwodem drukowanym)

B – moduł z anteną F-PCB (Antena z odwróconym obwodem drukowanym)

C – moduł z anteną ceramiczną

D – moduł z anteną dipolową

Następnie zbadali modulację, szybkość transmisji i siłę sygnału. Otrzymane wyniki przedstawia Tabela 3.3, z której wynika, że wraz ze wzrostem przepustowości wzrasta jakość sygnału.

Tablica 3.3: Parametry sygnału [19].

Modulacja	Szybkość transmisji	Siła sygnału
DSSS	1 Mbps	-98 dBm
CCK	11 Mbps	-91 dBm
$\frac{1}{2}$ BPSK	6 Mbps	-93 dBm
$\frac{3}{4}$ 64-QAM	54 Mbps	-75dBm
HT20, MCS7	72.2 Mbps	-72 dBm

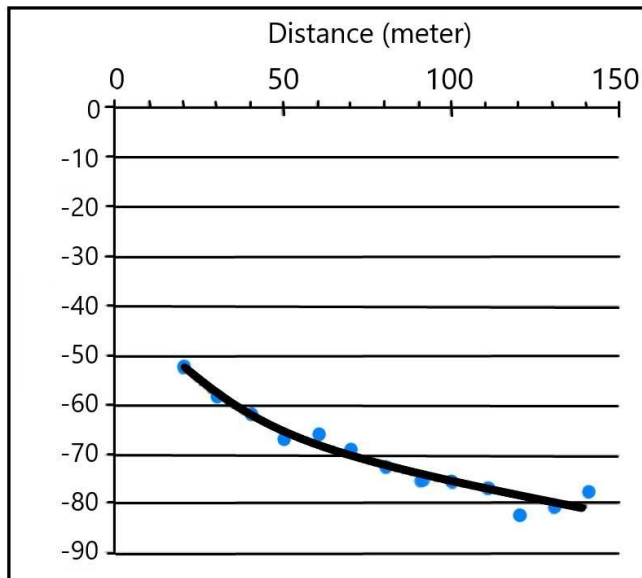
Opisywane badanie polegało na wysyłaniu sygnału pomiędzy dwoma modułami (jeden ustawiony jako stacja, drugi natomiast jako punkt dostępowy) na wysokości 63 cm nad ziemią w otwartej przestrzeni. Sygnał był wysyłany z pełną mocą, która wynosiła +17 dBm w standardzie 802.11g. Otrzymane wyniki przedstawia Tabela 3.4.

Bazując na otrzymanych wynikach możemy podczas projektowania sieci czujników odpowiednio dobrać moduł w zależności od odległości pomiędzy nadajnikiem a odbiornikiem z zachowaniem odpowiedniej siły sygnału. Jednakże najlepszy wynik uzyskał moduł z anteną ceramiczną i dodatkowo

Tablica 3.4: Średni wskaźnik siły odbieranego sygnału (RSSI) (dB) [19].

Odległość Model	20 m	30 m	40 m
A	-75,24	-79,19	-84,51
B	-79,75	-85,83	-90,94
C	-67,83	-73,75	-80,59
D	-52,23	-58,00	-61,38

gniazdem IPX RF (miniaturowe złącze antenowe na zatrzask) do podłączenia dodatkowej anteny dipolowej. Pomiar zasięgu skutecznego sygnału został przedstawiony na Rysunku 3.4.



Rysunek 3.4: Rozszerzony pomiar RSSI modelu D [19].

Z powyższego wykresu możemy odczytać zależność, iż wraz ze wzrostem dystansu, maleje jakość sygnału. Jednakże przy zastosowaniu anteny dipolowej skuteczny zasięg wzrasta do 140 m.

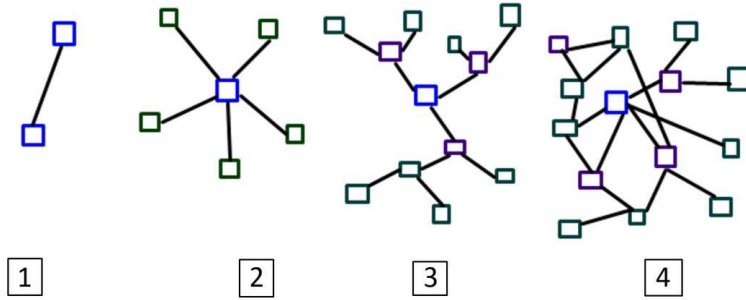
3.4. Wi-Fi – charakterystyka i topologie

Popularne Wi-Fi jest jednym ze standardów, służących do przekazywania danych. Znajduje zastosowanie w wielu dziedzinach naszego życia. Standard ten wykorzystywany jest w bezprzewodowej komunikacji o zasięgu lokalnym (LAN), dla węzłów o charakterze stałym lub poruszających się. Niewątpliwą zaletą tego standardu jest powszechność modemów zgodnych z IEEE (Institute of Electrical and Electronics Engineers) 802.11, wbudowanych w urządzenia konsumenckie takie jak komputery, tablety, telefony komórkowe, urządzenia RTV i AGD, itp. Charakteryzuje się on dużą szybkością transmisji danych. Standard ten definiuje podstawę kontroli dostępu do medium (MAC) oraz wiele warstw fizycznych (PHY), pracujących w zakresie częstotliwości 2.4 GHz oraz 5.0 GHz. Sieć Wi-Fi wykorzystuje fale radiowe o określonej częstotliwości do transmisji danych. Sieć ta posiada ograniczony zasięg i stałe źródło zasilania. Wspomniany moduł pracuje w standardzie 802.11 b/g/n.

Tablica 3.5: Charakterystyka sieci Wi-Fi pasma 802.11 b/g/n.

Pasma	Prędkość	Częstotliwość	Opis/charakterystyka
802.11b	11 Mb/s	2,4 GHz	Standard jest znany pod nazwą Wi-Fi (Wireless Fidelity), działa w pasmie częstotliwości 2,4 GHz.
802.11g	54 Mb/s	2,4 GHz	Wykorzystuje technologię ODM (orthogonal Frequency Division Multiplexing). Urządzenia mogą współpracować z urządzeniami w standardzie 802.11b.
802.11n	300 Mb/s	5 GHz	Standard posługuje się technologią MIMO (Multiple Input, Multiple Output), która wykorzystuje wiele kanałów transmisyjnych do stworzenia jednego połączenia
	150 Mb/s	2,4 GHz	

Istotnym elementem każdej sieci jest jej topologia, czyli sposób połączenia poszczególnych węzłów. Istnieje kilka podstawowych topologii, wykorzystywanych podczas budowania infrastruktury sieci przewodowej, jak i bezprzewodowej. Przedstawia je Rysunek 3.5.



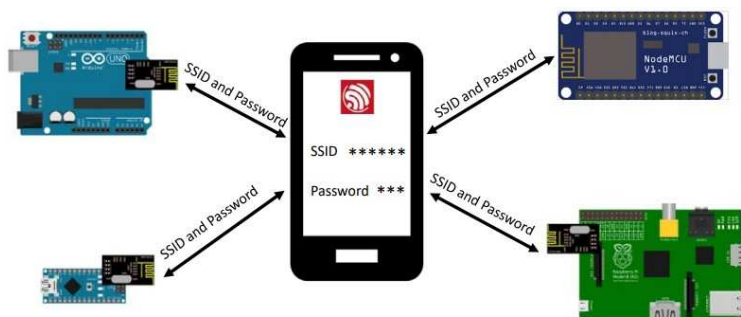
Rysunek 3.5: Topologie sieci bezprzewodowych: 1) punkt-punkt, 2) gwiazda, 3) drzewo, 4) siatka.

Topologia punkt-punkt (point- point) – umożliwia połączenie jedynie dwóch węzłów komunikujących się między sobą. Jeden z węzłów pełni rolę nadrzędną a drugi podrzędną. Węzeł nadrzędny jest koordynatorem sieci. Topologia ma zastosowanie w sieciach ad-hoc, gdzie węzeł podrzędny jest miernikiem/licznikiem a węzeł nadrzędny jest odpowiedzialny za akwizycje danych lub przekazywanie ich do nadrzędnego systemu.

Topologia gwiazdy (star) – jest rozwinięciem wspomnianej wyżej topologii. Sieć stworzona według tej struktury o dwustopniowej hierarchii, gdzie wszystkie węzły podrzędne komunikują się bezpośrednio z węzłem nadrzędnym.

Topologia drzewa (tree) – jest rozwinięciem topologii gwiazdy o dodatkowe stopnie hierarchii. Węzły podrzędne (liść, leaf) komunikują się z koordynatorem sieci (korzeń, root) za pośrednictwem węzłów pośrednich (gałąź, router).

Topologia siatki (mesh) – zakłada brak sztywnej hierarchii sieci. Każdy węzeł może komunikować się z innym węzłem, dane mogą być przekazywane między węzłami pośrednio, poprzez inne węzły ustalona ścieżką.



Rysunek 3.6: Topologie sieci bezprzewodowych [11].

Smartfon pozwala na podłączenie czujników do Wi-Fi. Min i wsp. w swojej pracy [11] opisali Technologię Smart-config, opartą na protokole ESP-TOUCH, a opracowaną przez firmę Espressif Systems. Technologia ta ma na celu wspieranie połączenia z routerem urządzeń Wi-Fi. Graficzny interfejs użytkownika (GUI), za pomocą smartfona lub dowolnego innego urządzenia z obsługą Wi-Fi nadaje pakiet UDP, którego treść zawiera zakodowany identyfikator SSID i hasło. Ten protokół komunikacyjny wspomaga moduł ESP8266 w pobieraniu i dekodowaniu danych. Istnieją trzy główne zalety zastosowania technologii smart-config:

- wygodne podejście do modułu ESP8266,
- dostępna konfiguracja Wi-Fi przy użyciu smartfona,
- możliwość całkowitego pominięcia aktualizacji kodu źródłowego i przesyłania oprogramowania układowego, co ogranicza koszty.

Jednoczesna ponowna konfiguracja kilku urządzeń może zaoszczędzić wiele wysiłku. Nie mniej ważnym, po udanym połączeniu z Wi-Fi, połączeniu SSID i hasła, jest zapisanie w EEPROM modułu ESP8266 do późniejszego wykorzystania. Ta metoda pomaga modułowi ESP8266 automatycznie łączyć z Wi-Fi po przerwie w zasilaniu, w innym przypadku wymagane jest nadawanie pakietu UDP.

Ponadto ESP8266 jest zaprojektowanym układem zapewniającym pełną łączność z Internetem.

Protokół kontroli transferu (TCP/IP) – stanowi zestaw standardowych protokołów używanych do ustanowienia połączenia pomiędzy urządzeniami

w sieci. TCP działa w trybie klient-serwer. Klient inicjuje połączenia a serwer oczekuje na nawiązanie połączenia na określonym porcie. TCP jest to kanał transmisji, w którym odbiorca potwierdza otrzymanie danych. W razie potrzeby dane są przesyłane ponownie. Jednakże tym zajmuje się sam protokół TCP i na koniec informuje aplikację, że transmisja zakończyła się sukcesem.

Protokół UDP nie posiada takiej funkcji jak TCP. Jego użycie nie zapewnia 100% gwarancji dostarczenia danych. Ten typ komunikacji jest szybszy niż TCP. Powodem wystąpienia możliwości częściowej utraty danych, jest brak potwierdzania otrzymanych danych podczas komunikacji. UDP stosuje się w miejscach, gdzie prędkość jest wyższym priorytetem niż dokładność, np. w wideokonferencjach. Jeśli tam niektóre piksele nie zostaną przesłane, nie wpłynie to znacznie na jakość wideo, za którą w głównej mierze odpowiada prędkość transferu.

Większość projektów i kodów ESP8266 działa w sieci TCP/IP, UDP. Istotną zaletą modułu ESP8266 jest możliwość pracy jako punktu dostępu, zapewniając połączenie internetowe z innymi urządzeniami (zwykle nazywane stacją). Punkt dostępowy można użyć do utworzenia małej sieci kratowej. Jednakże liczba jednocześnie podłączonych urządzeń ograniczona jest do pięciu. Oczywiście skonfigurowany moduł ESP w trybie stacji, można podłączyć do punktu dostępowego za pomocą innego modułu ESP.

3.5. Charakterystyka bezprzewodowych sieci czujników i zastosowanie

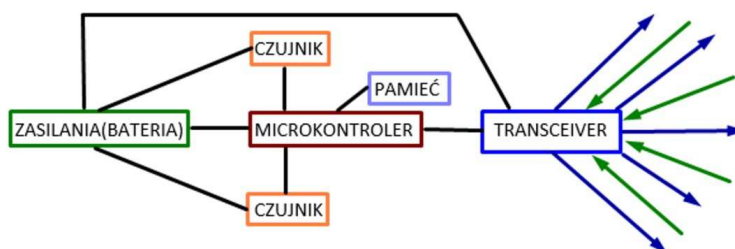
W obecnych czasach zastosowanie bezprzewodowych czujników obejmuje praktycznie każdą gałąź przemysłu, jak również każdą dziedzinę życia na całym świecie. Od dawien dawna człowiek chciał uzyskiwać informacje na interesujący go temat, jak również nie będąc na danym obszarze kontrolować/odczytywać interesujące go parametry. W dobie dzisiejszej komunikacji bezprzewodowej stało się to możliwe i wkroczyło w każdą strefę naszego życia.

Tak naprawdę mierzyć można wszystko i wszędzie używając do tego odpowiednich urządzeń zwanych czujnikami. Zasada działania czujników najczęściej sprowadza się do zmiany wielkości fizycznych na elektryczne, możliwe do pomiaru i dalszego przetworzenia na np. na postać cyfrową.

Możemy wyróżnić kilka podstawowych i najczęściej spotykanych czujników, takich jak: czujnik pomiaru temperatury, wilgotności, ciśnienia, wysokości, siły nacisku, koloru, ruchu, obrotów, dymu, natężenia światła, wibracji, przyspieszenia, składu chemicznego, zapylenia itd.

Bezprzewodowe sieci czujników zbudowane są z rozproszonych autonomicznych urządzeń, integrujących w sobie urządzenie pomiarowe z nadajnikiem i odbiornikiem radiowym. Parametry mierzone w sieciach sensorycznych to np. temperatura, ciśnienie, wilgotność, wibracje, przyspieszenie, skład chemiczny, zapylenie itp. Otrzymane dane pomiarowe mogą być wstępnie przetworzone przez węzeł sieci, a następnie przekazane drogą radiową do węzła centralnego, gdzie zostaną przetworzone po wcześniejszej obróbce. Tego typu sieci nazywamy bezprzewodowymi sieciami sensorycznymi WSN – Wireless Sensor Network. Wyróżnia się ponadto sieci WSAAN – Wireless Sensor and Actuators Network. W sieciach tego typu zakłada się również elementy wykonawcze (np. pompa, siłownik).

Budowę węzła bezprzewodowej sieci prezentuje poniższy schemat (Rysunek 3.7).



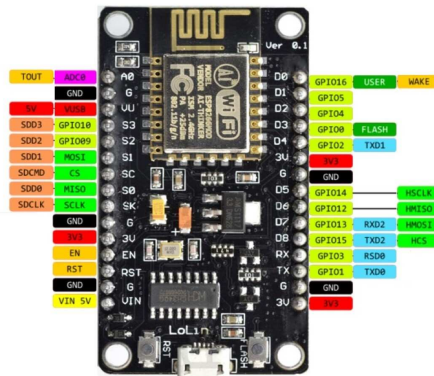
Rysunek 3.7: Schemat węzła bezprzewodowej sieci czujników.

Węzeł sieci sensorycznej może posiadać jeden lub wiele czujników, odcytujących określony parametr. Sygnał elektryczny jest zamieniany w postać cyfrową i przekazywany wewnątrz sieci, zgodnie z przyjętym protokołem transmisji. Za wysyłanie i odbiór danych odpowiedzialny jest układ transceivera. Natomiast za obsługę pomiaru i przekazywania danych odpowiedzialny jest mikrokontroler, który stanowi serce całego modułu. Obecnie najczęściej stosowane pasma częstotliwości to 2.4 GHz i 5 GHz. Jednakże gama używanych częstotliwości jest znacznie szersza.

Węzeł bezprzewodowej sieci wymaga autonomicznego źródła zasilania. Najczęściej jest to bateria. Jej pojemność jest znaczącym elementem decydującym o życiu węzła. W zależności od pojemności źródła zasilania, moduł może działać od kilku godzin do nawet kilku lat. Współczesne rozwiązania proponują możliwości uzupełniania energii elektrycznej, pozyskując ją z otoczenia poprzez zastosowanie różnego rodzaju przetworników konwertujących energię kinetyczną, słoneczną lub ciepłą na energię elektryczną. Jednakże kluczową sprawą jest minimalizacja poboru energii, poprzez zastosowanie odpowiedniej konstrukcji sprzętowej, dobór algorytmów sterujących i pomiarowych oraz zastosowanie odpowiednich protokołów komunikacyjnych.

3.6. Budowa przykładowej sieci Wi-Fi z zastosowaniem modułu ESP8266

Do budowy przykładowej sieci został użyty moduł płytki rozwojowej NodeMCU V3. To platforma IoT typu open source, która zawiera oprogramowanie układowe działające na układzie ESP8266 WI-FI firmy Espressif systems. Zastosowano w nim moduł ESP-12. Nazwa NodeMCU odnosi się do zastosowanego oprogramowania układowego. Oprogramowanie wykorzystuje język skryptowy Lua. Zastosowane interaktywne oprogramowanie dla modułów Expressif ESP8622 zawiera układ CP2102 TTL oraz USB do programowania i debugowania. Jest on przyjazny dla płytki prototypowej i może być zasilany przez port micro USB. Poszczególne piny zostały scharakteryzowane w Tabeli 3.6, a ich umiejscowienie przedstawia Rysunek 3.8.

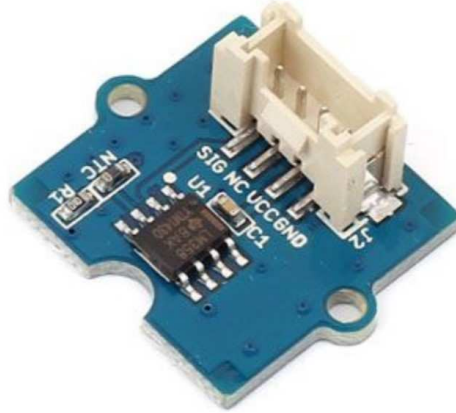


Rysunek 3.8: Opis pinów NodeMecu V3.

Tablica 3.6: Opis poszczególnych pinów NadeMcu V3.

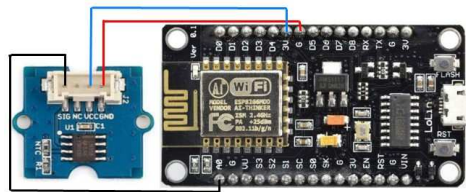
Nazwa Pin PIN	Alternatywna nazwa	Funkcja (opis skrótu)
TX	TXD	Nadawanie na porcie szeregowym
RX	RDX	Odbiór
A0	A0	Wejście analogowe max 3,3 V
D0	GPIO16	IO budzenie się ze stanu uśpienia
D1	GPIO5	IO, SCL (Serial Clock Line)
D2	GPIO4	IO, SDA (Serial Data Line)
D3	GPIO0	IO, 10k Pull-up
D4	GPIO2	IO, 10k Pull-up, BUILTIN_LED
D5	GPIO14	IO, SCK
D6	GPIO12	IO, MISO
D7	GPIO13	IO, MOSI
D8	GPIO15	IO, 10k Pull-down, SS
G	GND	Ground
5V0	5,0 V	Napięcie 5,0 V
3V3	3,3 V	Napięcie 3,3 V
RST	RTS	Reset

Do budowy sieci użyto czujnika temperatury z Zestawu ARDUINO GROVE. Czujnik ten do pomiaru temperatur wykorzystuje termistor. Rezystancja termistora zmienia się w zależności od temperatury otoczenia. Wartość tej rezystancji zmienia wyjście dzielnika napięcia, które jest mierzone przez analogowy styk wejściowy i przekształca na wartość temperatury. Zakres pracy czujnika wynosi od -40 do 125°C z dokładnością do 1.5°C . Na Rysunku 3.9 przedstawiono omawiany czujnik.



Rysunek 3.9: Czujnik temperatury (Termistor NCP18WF104F03RC). Źródło: <https://www.seedstudio.com/Grove-Temperature-Sensor.html>.

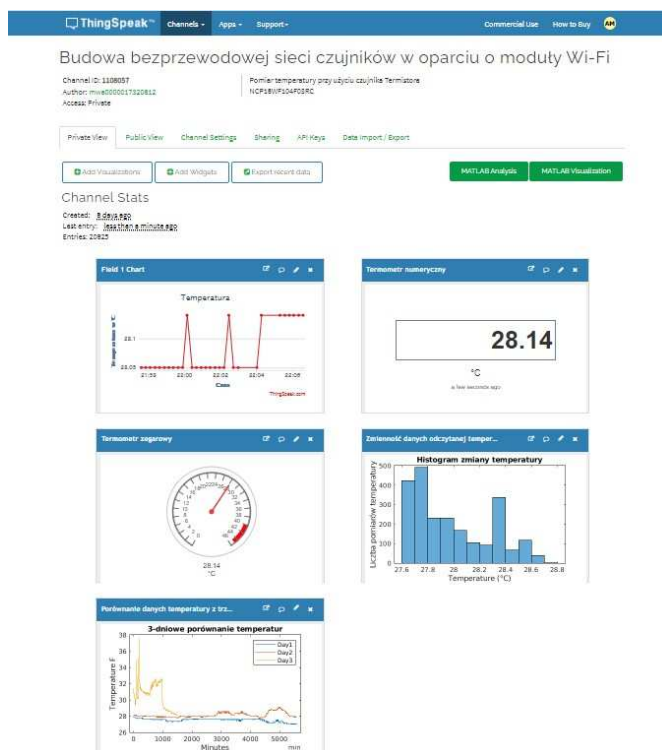
Na Rysunku 3.10 przedstawiono schemat prezentujący sposób podłączenia czujnika z modulem NodeMcu V3. Piny GND połączono czerwonym przewodem. Napięcie z modułu 3.3 V podłączono do pinu VCC w czujniku, natomiast pin sygnałowy modułu z pinem analogowym A0. Pinu analogowego (A0) nie należy łączyć z czujnikiem zasilanym napięciem większym od 3.3 V.



Rysunek 3.10: Schemat podłączenia czujnika do modułu NodeMcu V3.

Dane odczytywane przez czujnik są przetwarzane i w formie cyfrowej przesyłane przy pomocy ESP8266 do platformy ThingSpeak. Jest to jedna z platform świadczących usługi odczytywania i analizowania danych w czasie rzeczywistym do projektów IoT. Dane możemy monitorować, pod warunkiem posiadania dostępu do Internetu. ThingSpeak wyposażona jest w analitykę MATLABa, która pozwala na zbieranie, wizualizację oraz analizę danych w chmurze. ThingSpeak udostępnia tworzenie i wykonywanie programów w kodzie MATLAB, co umożliwia przeprowadzenie wstępnej

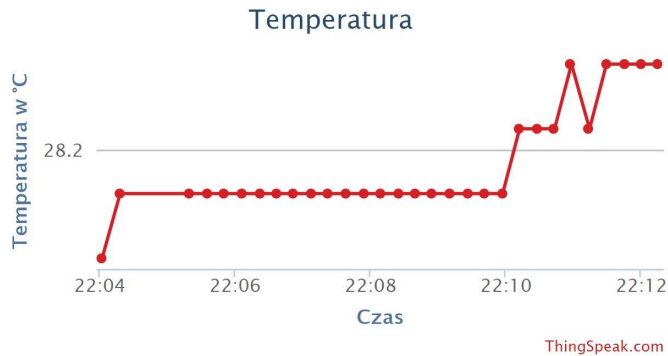
obróbki analizy i wizualizacji danych. Na Rysunkach 3.11, 3.12, 3.13, 3.14, 3.15 przedstawiono wizualizacje danych przesłanych przez użyty w eksperymencie moduł (<https://thingspeak.com/>).



Rysunek 3.11: Platforma ThingSpeak z wizualizacją odczytywanych danych.

W celu umożliwienia przesłania danych na początku trzeba założyć konto na ThingSpeak.com. Kolejny etap stanowi zalogowanie się i utworzenie kanału komunikacyjnego. Menu programu jest proste i intuicyjne. Uzyskanie dostępu do platformy umożliwia wygenerowanie klucza dostępu (klucz API zapisu i odczytu). Następnie należy utworzyć miejsce, gdzie będą przesyłane dane w celu analizy jak również archiwizacji. Do tego zadania używa się Platformy IoT oraz arkusza kalkulacyjnego do archiwizacji. Arkusz trzeba połączyć z aplikacją internetową, która będzie komunikowała się z modułem NodeMecu. Za sterowanie modułu i przebieg całego procesu odpowiada kod zapisany w pamięci modułu. Treść kodu wraz z opisem podstawowych danych przedstawia Listing 3.6 oraz 3.6. Potwierdzenie przesłanych danych w interwale czasowym prezentuje Listing 3.6. Z kolei na

Rysunku 3.17 pokazano zastosowany do eksperymentu moduł wraz z czujnikiem.

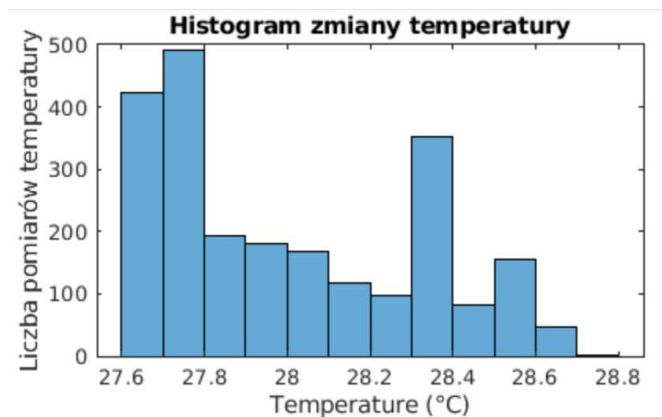


Rysunek 3.12: Wyeksportowany wykres temperatury w formie graficznej.

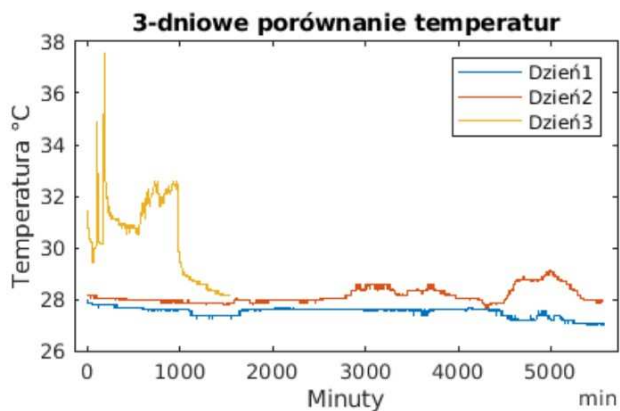
Dane można również przedstawić w formie jednego z kilku możliwych widżetów.



Rysunek 3.13: Przedstawienie temperatury w formie termometru tarczowego.



Rysunek 3.14: Wykres przedstawiony przy użyciu jednej z opcji programu MATLAB Analysis.



Rysunek 3.15: Wizualizacja danych przy użyciu Programu MATLAB Visualization.

Poniżej przedstawiono kod programu.

Listing 3.1: Kod programu odpowiedzialny za uruchomienie i połączenie WI-FI.

```
1 #include <ESP8266WiFi.h>
2
3 String apiWritekey = "xxxxxxxxxxxxxxxxxxxx";
4 const char *ssid = "xxxxxxxx";
5 const char *password = "xxxxxxxx";
6
7 const int pinTemp = A0;
8 float temperatura;
9 int B=3975;
10 float resistance;
11
12 const char *server = "api.thingspeak.com";
13 float resolution = 3.3/1023;
14 WiFiClient client;
15
16 void setup()
17 {
18   Serial.begin(115200);
19   WiFi.disconnect();
20   delay(10);
21   WiFi.begin(ssid, password);
22   Serial.println();
23   Serial.println();
24   Serial.print("Łączenie z");
25   Serial.println(ssid);
26
27   WiFi.begin(ssid, password);
28
29   while (WiFi.status() != WL_CONNECTED) {
30     delay(500);
31     Serial.print(".");
32   }
33
34   Serial.println("");
35   Serial.print("NodeMcu połączony z WI-FI");
36   Serial.println(ssid);
37   Serial.println();
38 }
```

Listing 3.2: Wyjaśnienie dotyczące informacji zawartych w kodzie programu rozpoczęto od zastosowanej biblioteki.

```
1 #include <ESP8266WiFi.h> //biblioteka zapewniająca wykonanie procedur
   Wi-Fi specyficznych dla ESP8266.
2
3 String apiWritekey = "Klucz zapisu API"; // kod ten otrzymuje się
   podczas konfiguracji konta na platformie
4
5 const char *ssid = "Nazwa Sieci Wi-Fi, do której zamierzamy się włą
   czyć";
6
7 const char *password = "Hasło dostępu do sieci Wi-Fi";
```

Powyższy kod odpowiedzialny jest za uruchomienie, połączenia Wi-Fi.

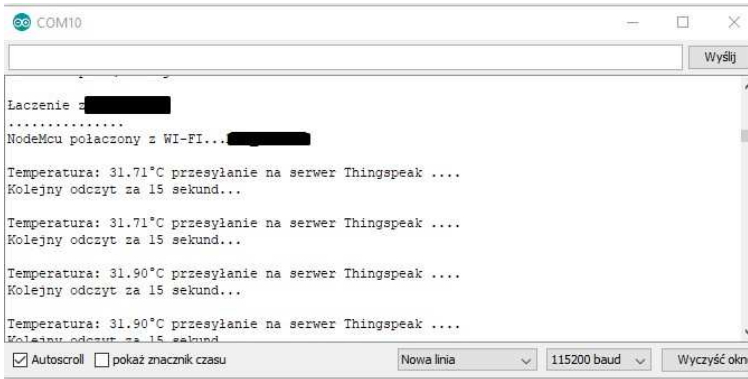
Listing 3.3: Kod programu odpowiedzialny za przesyłanie danych na serwer ThingSpeak.

```
1 void loop()
2 {
3   int val = analogRead(pinTemp);
4   resistance=(float)(1023-val)*10000/val;
5   temperatura=1/(log(resistance/10000)/B+1/298.15) -273.15;
6
7   if (client.connect(server,80)) {
8     String Dane = apiWritekey;
9     Dane += "&field1=";
10    Dane += String(temperatura);
11    Dane += "\r\n\r\n";
12
13    client.print("POST /update HTTP/1.1\n");
14    client.print("Host: api.thingspeak.com\n");
15    client.print("Connection: close\n");
16    client.print("X-THINGSPEAKAPIKEY: "+apiWritekey+"\n");
17    client.print("Content -Type: application/x-www-form-urlencoded\n")
18    ;
19    client.print("Content -Length: ");
20    client.print(Dane.length());
21    client.print("\n\n");
22    client.print(Dane);
23
24    Serial.print("Temperatura: ");
25    Serial.print(temperatura);
26    Serial.println("stp. C przesyłanie na serwer Thingspeak ....");
27  }
28
29  client.stop();
30
31  Serial.println("Kolejny odczyt za 15 sekund...");
32  Serial.println();
33  delay(15000);
34 }
```

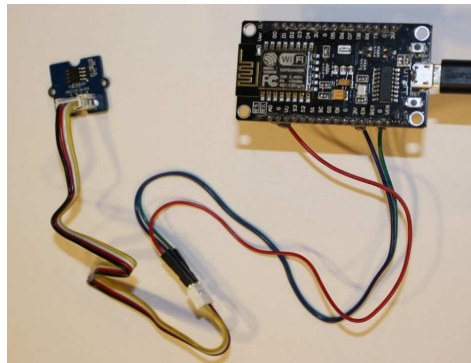
Zacytowany kod programu odpowiedzialny jest za przesyłanie danych

na serwer ThingSpeak.

W kolejnym etapie umieszczamy kod w funkcji „void loop”, aby wielokrotnie wykonywał polecenia. W tej części kodu umieszczamy również informacje dotyczące podłączonego czujnika temperatury. Zastosowanie instrukcji „if” pozwala na połączenie się z serwerem ThingSpeak, i przy użyciu modułu ESP8266 przesłanie aktualnej wartości temperatury w określonym interwale czasowym.



Rysunek 3.16: Zapis przedstawiający przysyłanie i potwierdzenie otrzymanej informacji w interwale czasowym.



Rysunek 3.17: Rzeczywiste połączenie modułu z czujnikiem zastosowanego w eksperymencie.

3.7. Podsumowanie

Podsumowując należy stwierdzić, że sieci bezprzewodowe ulegają ciągłym modyfikacjom i dynamicznym przemianom. Zauważalne jest stałe poprawianie i tworzenie nowych, sprawniejszych, szybszych i wymagających mniejszych nakładów energii protokołów przekazywania danych. Zmiany te wymuszane są przez konieczność szybkiego dostarczania informacji w zglobalizowanym świecie a także zmieniający się tryb życia człowieka. W obecnym czasie - światowej pandemii mogliśmy doświadczyć dobrodziejstwa sieci bezprzewodowych. Dzięki zastosowaniu WSE możemy monitorować interesujące nas parametry w wielu miejscach jednocześnie bez zbędnego przemieszczania. Powstaje coraz więcej platform internetowych, które zajmują się analizowaniem, jak również przechowywaniem i archiwizowaniem otrzymanych danych. Dane te po przesłaniu przez klienta mogą być wykorzystywane z zachowaniem zasad bezpieczeństwa. Odczytywać i administrować danymi możemy z każdego miejsca na ziemi używając do tego elektronicznych urządzeń komunikacyjnych. Jesteśmy świadkami potężnego wzrostu komunikacji informacyjnej w każdej dziedzinie życia.

Bibliografia

- [1] S.I. Abdullahi, M.H. Habaebi, N.A. Malik. Flood Disaster Warning System on the go. *7th International Conference on Computer and Communication Engineering (ICCCE)*, strony 258–263, 2018.
- [2] N. Al-Atwan, M. Nitulescu. Wireless control system for an intelligent home. *20TH INTERNATIONAL CARPATHIAN CONTROL CONFERENCE (ICCC)*, strony 330–335. IEEE, 2019.
- [3] Espressif Inc. *ESP8266EX Datasheet Version 6.5*, 2020.
- [4] R. Khamitkar, F. Valsangkar. Design and Implementation of Internet of Things Based Multi-sensor Device. *Computing, Communication and Signal Processing*, strony 157–163, 2019.
- [5] K. Kodali, S. Soratkal. MQTT based Home Automation System Using ESP8266. *IEEE REGION 10 HUMANITARIAN TECHNOLOGY CONFERENCE (R10-HTC)*, IEEE Region 10 Humanitarian Technology Conference, strony 784–791. IEEE, 2016.
- [6] R.K. Kodali, A. Valdas. MQTT Based Environment Monitoring In Factories for Employee Safety. *PROCEEDINGS OF THE 2017 3RD*

- INTERNATIONAL CONFERENCE ON APPLIED AND THEORETICAL COMPUTING AND COMMUNICATION TECHNOLOGY (ICATCCT)*, strony 152–155. IEEE, 2017.
- [7] Z. Li, J. Wang, R. Higgs, L. Zhou, W. Yuan. Design of an Intelligent Management System for Agricultural Greenhouses based on the Internet of Things. *International Conference on Computational Science and Engineering*, strony 154–160. IEEE, 2017.
- [8] M. Malhotra, I.K. Aulakh, N. Kau, N.S. Aulakh. Air Pollution Monitoring Through Arduino Uno. *ICT Systems and Sustainability*, strony 235–243, 2020.
- [9] J. Mesquita, D. Guimaraes, C. Pereira, F. Santos, L. Almeida. Assessing the ESP8266 WiFi module for the Internet of Things. *23RD INTERNATIONAL CONFERENCE ON EMERGING TECHNOLOGIES AND FACTORY AUTOMATION (ETF A)*, IEEE International Conference on Emerging Technologies and Factory Automation-ETF A, strony 784–791. IEEE, 2018.
- [10] D. Minchev, A. Dimitrov. Home automation system based on ESP8266. *20TH INTERNATIONAL SYMPOSIUM ON ELECTRICAL APPARATUS AND TECHNOLOGIES (SIELA)*, International Symposium on Electrical Apparatus and Technologies. IEEE, 2018.
- [11] A.T.T. Minh, N.L. Trong, P.V. Tan. Smart-config wifi technology using esp8266 for low-cost wireless sensor networks. *INTERNATIONAL CONFERENCE ON ADVANCED COMPUTING AND APPLICATIONS (ACOMP)*, strony 22–28. IEEE, 2018.
- [12] F.O. Ovalles, A.E. Bolivar, A.J. Rodriguez. Use of an embedded system with wifi technology for domotic control of conventional environments. *INTERNATIONAL MEETING ON APPLIED SCIENCES AND ENGINEERING*, wolumen 1126 serii *Journal of Physics Conference Series*. IOP, 2018.
- [13] I. Sarkar, B. Pal, A. Datta, R. Sandip. Wi-Fi-Based Portable Weather Station for Monitoring Temperature, Relative Humidity, Pressure, Precipitation, Wind Speed, and Direction. *Information and Communication Technology for Sustainable Development*, strony 399–404, 2020.
- [14] M. Schwartz. *Internet of Things with ESP8266*. Published by Packt Publishing Ltd, 2016.

- [15] K.S. Shinde, P.H. Bhagat. Industrial Process Monitoring Using IoT. *INTERNATIONAL CONFERENCE ON I-SMAC (IOT IN SOCIAL, MOBILE, ANALYTICS AND CLOUD) (I-SMAC)*, strony 39–42. IEEE, 2017.
- [16] W.G. Shun, W.M.W. Muda, W.H.W. Hassan, A.Z. Annua. Wireless Sensor Network for Temperature and Humidity Monitoring Systems Based on NodeMCU ESP8266. *Advances in Cyber Security*, strony 262–273, 2020.
- [17] S. Thakare, P.H. Bhagat. Arduino-Based Smart Irrigation Using Sensors and ESP8266 WiFi Module. *PROCEEDINGS OF THE 2018 SECOND INTERNATIONAL CONFERENCE ON INTELLIGENT COMPUTING AND CONTROL SYSTEMS (ICICCS)*, strony 1085–1089. IEEE, 2018.
- [18] Z. Wan, Y. Song, Z. Cao. Environment Dynamic Monitoring and Remote Control of Greenhouse with ESP8266 NodeMCU. *PROCEEDINGS OF 2019 IEEE 3RD INFORMATION TECHNOLOGY, NETWORKING, ELECTRONIC AND AUTOMATION CONTROL CONFERENCE (ITNEC 2019)*, strony 377–382. IEEE, 2019.
- [19] Yoppy, R.H. Arjadi, H. Candra, H.D. Prananto, T.A.W. Wijanarko. RSSI Comparison of ESP8266 Modules. *ELECTRICAL POWER, ELECTRONICS, COMMUNICATIONS, CONTROLS, AND INFORMATICS SEMINAR (EECCIS)*, strony 150–153. IEEE, 2018.
- [20] F. Zhang, M. Liu, Z. Zhou, W. Shen. An IoT-Based Online Monitoring System for Continuous Steel Casting. *IEEE INTERNET OF THINGS JOURNAL*, 3(6):1355–1363, 2016.

Building of the wireless sensor network based on WI-FI modules

Abstract: Currently, we notice a stable upward trend worldwide as far as the number of devices with wireless communication is concerned. As a result, there is a robust and versatile use of Internet based items (IoT). So called Wireless Sensor Network (WSN) entails a group of spread out sensors linked to the same wireless network. One representative example with an increasing popularity is module ESP8266 produced by Chinese company Espressif. Thanks to its features and tiny dimensions as well low energy consumption the module has its application in many areas of life such as irrigation, agriculture, weather elements, but first and foremost in the management of production and technological processes. The

environmental data is visually arrayed so as to enable user to control and manage remotely, even using a smartphone. Use of devices, which monitor and control through the existing Wi-Fi infrastructure can reduce significantly costs of production with the same level of efficiency resulting in the positive impact on many sectors of the economy as well as environmental protection. The article reviews briefly main features of the circuit board with the module ESP8266 (NodeMecu) and its practical application in monitoring temperature with transfer of data to Thing Speak platform where this is processed further.

Spis rysunków

1.1	Elementy sterujące w szafie elektrycznej – część zarządzająca.	18
1.2	Elementy sterujące w szafie elektrycznej – część wykonawcza.	19
1.3	Pneumatyczne wyspy zaworowe z interfejsem sieciowym. . .	19
1.4	Falownik sieciowy silnika transportera.	20
1.5	Switche zarządzalne („managed”) sieci wysokiego poziomu.	22
1.6	Topologia sieci Ethernet z rozdziałem na dwa poziomy (lo- giczny i wykonawczy).	22
1.7	Punkt podziału sieci wysokiego i niskiego poziomu.	23
1.8	Schemat sieci przemysłowej AS-i.	23
1.9	System monitoringu pracy zakładu.	25
1.10	Ekran główny systemu monitoringu.	26
1.11	Ekran logowanych danych predykcyjnych.	27
1.12	Logika obsługi ostrzeżenia, przekroczenia parametru granicz- nego silnika serwo w programie drabinkowym RSLogix 5000, sterownika PLC.	28
1.13	Komunikat ostrzeżenia, wyświetlany na ekranie pulpitu ope- ratorskiego – definicja komunikatu w programie Factory Talk View Studio.	29
1.14	Logika ostrzeżenia, przekroczenia parametru granicznego ok- na impulsów w programie drabinkowym RSLogix 5000, ste- rownika PLC.	30
1.15	Komunikat ostrzeżenia na ekranie pulpitu operatorskiego – informacja do operatora, o konieczności wezwania służb UR.	31
1.16	Monitoring wibracji łożysk silnika, skorelowany z poborem energii.	32
1.17	Logika w programie drabinkowym, sterownika PLC dla in- formacji ostrzegawczej o przekroczeniu wibracji.	33
1.18	Komunikat ostrzeżenia na ekranie pulpitu operatorskiego – informacja do operatora, o konieczności zmiany prędkości wentylatorów celem redukcji wibracji.	33

1.19	Monitoring wartości ciśnień.	34
1.20	Wstępna definicja modelu maszyny do predykcji poziomu technicznego downtime'u.	36
1.21	Wynik regresji dla jednego z wyjść modelu.	36
1.22	Wynik modelowania jako równanie matematyczne.	37
1.23	Zredefiniowany model maszyny.	37
1.24	Przykłady usprawnień na maszynach.	39
2.1	Schemat sieci czujników bezprzewodowych w topologii kra- towej.	48
2.2	Schemat sieci czujników bezprzewodowych w topologii gwia- zdy.	49
2.3	Schemat sieci czujników bezprzewodowych w topologii drzewa.	50
2.4	Wizualizacja przepływu danych pomiędzy węzłami w ekspe- rymentalnej sieci bezprzewodowej pozbawionej punktu cen- tralnego.	55
2.5	Moduł RFM69 - wyprowadzenia sygnałów modułu.	57
2.6	Rodzaje cyfrowej modulacji sygnału w module RFM69 w iden- tyfikacji stanów logicznych.	59
2.7	Struktura pakietu z podziałem na część dodawaną automa- tycznie przez transceiver (3 bajtowa preambuła, 2 bajty syn- chronizacyjne, konfiguracja transmisji CTL, suma kontrolna pakietu CRC) oraz bibliotekę obsługi (rozmiar, adres urzą- dzenia docelowego i źródłowego, oraz komunikat).	61
2.8	Interfejs SPI - transmisja danych.	62
2.9	Moteino – schemat modułu pomiarowego.	63
2.10	Widok zmontowanego modułu.	64
2.11	Moteino - wyprowadzenia modułu.	64
2.12	Pojedynczy moduł pomiarowy.	65
2.13	Zestaw pomiarowy moduły czujników i dodatkowy akwizytor danych.	65
2.14	Antena helikalna zamontowana wewnątrz czujnika.	68
2.15	Oprogramowanie komputerowe do akwizycji sieci czujników.	72
2.16	Wykres pomiaru zasięgu pomiędzy czujnikami.	74
2.17	Wykres pomiaru zasięgu pomiędzy czujnikiem a modułem akwizycyjnym.	74
3.1	Schemat blokowy ESP8266.	84
3.2	Wersje modułu ESP8266.	85
3.3	Opis podłączeń wyjść z modułu ESP8266 14E.	86

3.4	Rozszerzony pomiar RSSI modelu D.	88
3.5	Topologie sieci bezprzewodowych.	90
3.6	Topologie sieci bezprzewodowych.	91
3.7	Schemat węzła bezprzewodowej sieci czujników.	93
3.8	Opis pinów NodeMecu V3.	94
3.9	Czujnik temperatury (Termistor NCP18WF104F03RC). . .	96
3.10	Schemat podłączenia czujnika do modułu NodeMcu V3. . .	96
3.11	Platforma ThingSpeak z wizualizacją odczytywanych danych.	97
3.12	Wyeksportowany wykres temperatury w formie graficznej. .	98
3.13	Przedstawienie temperatury w formie termometru tarczowego.	98
3.14	Wykres przedstawiony przy użyciu jednej z opcji programu MATLAB Analysis.	99
3.15	Wizualizacja danych przy użyciu Programu MATLAB Visu- alization.	99
3.16	Zapis przedstawiający przysyłanie i potwierdzanie otrzy- mianej informacji w interwale czasowym.	102
3.17	Rzeczywiste połączenie modułu z czujnikiem zastosowanego w eksperymencie.	102

Spis tablic

2.1	Wyprowadzenia pinów modułu - nota katalogowa.	58
2.2	Kluczowe parametry modułu RFM69.	58
2.3	Podłączenie RFM69 do mikrokontrolera.	60
2.4	Dane techniczne zastosowanych anten helikalnych.	68
3.1	Zestawienie podstawowych wersji modułu ESP8266.	85
3.2	Opis poszczególnych wyjść z modułu ESP8266 14E.	86
3.3	Parametry sygnału.	87
3.4	Średni wskaźnik siły odbieranego sygnału (RSSI) (dB).	88
3.5	Charakterystyka sieci Wi-Fi pasma 802.11 b/g/n.	89
3.6	Opis poszczególnych pinów NadeMcu V3.	95