
	Instytut Informatyki Politechniki Śląskiej Zespół Mikroinformatyki i Teorii Automatów Cyfrowych			
Rok akademicki:	Rodzaj studiów*: SSI/NSI/NSM	Przedmiot (Języki Asemblerowe/SMiW):	Grupa	Sekcja
2022/2023	SSI	SMiW	2	1
Imię:	Dominik	Prowadzący: OA/JP/KT/GD/BSz/GB	JP	
Nazwisko:	Kijak			
<h2><i>Analiza Projektu</i></h2>				
<p>Temat projektu:</p> <h1 style="text-align: center;">Czujnik listów w skrzynce pocztowej</h1>				
Data oddania: dd/mm/rrrr		26.04.2023		

1. Cel zadania

Urządzenie powinno powiadomić użytkownika o dostarczeniu listu lub przesyłki przez email. W idealnym przypadku urządzenie byłoby bardzo łatwe w montażu, pasujące do każdego rozmiaru skrzynki, wykrywające każdy obiekt wrzucony do skrzynki oraz energooszczędne i wytrzymałe. Spełnienie wszystkich tych wymagań nie jest łatwe, więc analizę zacznę od wylistowania najważniejszych cech urządzenia, do których będę się później odnosił:

- dopasowanie do różnych rozmiarów skrzynki
- wykrywanie różnego rodzaju obiektów
- mały rozmiar
- łatwy montaż
- energooszczędność
- wytrzymałość
- praca w szerokim zakresie temperatury

2. Analiza zadania

2.1 Wybór rodzaju czujnika

Pierwszym krokiem będzie ustalenie jakiego rodzaju czujnik nada się do tego projektu najlepiej. Zacząłem od zaznajomienia się z rodzajami czujników na rynku, które mogą być wykorzystane w projekcie. Najbardziej obiecującymi są czujniki:

- krańcowe
- przerwania wiązki IR
- odległości (np. typu ToF)

Zaczynając od początku listy, czujnik krańcowy byłby dobrym rozwiązaniem ze względu na energooszczędność i precyzję w wykrywaniu listów, ale nie nada się do wykorzystania, ponieważ trzeba by dobrać jego rozmiar do rozmiaru skrzynki oraz musiałby być trwale do niej przymocowany, a chciałbym uniknąć wiercenia tak samo jak jakichkolwiek innych form niszczenia skrzynki pocztowej.

Czujniki przerwania wiązki podczerwieni mogą bardzo precyzyjnie wykrywać listy, ale montaż byłby problematyczny. Każda para nadajnika i odbiornika musiałaby być montowana przez użytkownika, a same czujniki byłyby połączone z urządzeniem przewodem, którego długości nie da się przewidzieć ze względu na różny rozmiar skrzynek. Byłoby to uciążliwe i wyglądało nieestetycznie po zamontowaniu.

Czujniki odległości, w tym ultradźwiękowe i podczerwone, w przeciwieństwie do krańcowych, nie wymagają kontaktu z listami co rozwiązuje niektóre problemy. Montowane są też pod jedną obudową urządzenia, bez potrzeby przeciągania dodatkowych przewodów. Przy odpowiednim programie potrafią być energooszczędne. Można tak skonfigurować program, by wykrywał za pomocą czujników wymiary skrzynki i stosując przerwania co jakiś czas sprawdzał czy czas odbicia jest taki sam, dzięki czemu urządzenie będzie pasowało do skrzynek o różnych wymiarach.

Czujniki odległościowe spełniają wszystkie najważniejsze wymagania i są najlepszym rozwiązaniem w tym projekcie, jednak pozostaje pytanie czy wybrać czujnik podczerwieni czy ultradźwięku?

Czujniki ultradźwiękowe mają zwykle szerszy kąt pomiaru, co na pewno ułatwi wykrywanie listów, ale mają też swoje wady. Przede wszystkim trzeba zwrócić uwagę na to, w jakich warunkach pracować będzie urządzenie. Można założyć, że w skrzynce nie będzie unoszącego się kurzu, ani innych czynników mogących zakłócić pracę urządzenia, za wyjątkiem panującej temperatury. Praca czujnika wykorzystującego podczerwień nie będzie zakłócona przez temperaturę, ale inaczej może to wyglądać w przypadku czujnika ultradźwiękowego. Chciałbym żeby urządzenie mogło pracować w temperaturach przynajmniej od $-10\text{ }^{\circ}\text{C}$ do $30\text{ }^{\circ}\text{C}$. W tak dużej różnicy temperatur trzeba wziąć pod uwagę prędkość dźwięku w powietrzu. By porównać teoretyczną różnicę w pomiarach przygotowałem poniższą tabelę.

stopnie Celsjusza	stopnie Kelwina	prędkość dźwięku [m/s]	obliczona odległość [cm] dla:			
			100us	150us	200us	300us
-10	263	324,88	3,25	4,87	6,50	9,75
0	273	331,00	3,31	4,97	6,62	9,93
10	283	337,01	3,37	5,06	6,74	10,11
20	293	342,91	3,43	5,14	6,86	10,29
30	303	348,71	3,49	5,23	6,97	10,46

różnica [cm] pomiędzy -10, a 30 stopni:	0,24	0,36	0,48	0,71
---	------	------	------	------

Tabela 1. Różnica odległości obliczonej względem prędkości dźwięku w różnych temperaturach

Prędkość dźwięku została obliczona wzorem:

$$v = 331 \frac{m}{s} * \sqrt{\frac{T}{273 K}}$$

Zakładając że głębokość skrzynki nie będzie przekraczać około 10 cm, to maksymalna różnica w obliczonej odległości może wynosić ~0,7 cm.

Mimo że trzeba wziąć ten błąd pod uwagę, to myślę że da się wykorzystać czujnik ultradźwiękowy bez mierzenia temperatury otoczenia.

Jednak temperatura nie jest jedynym czynnikiem mającym wpływ na precyzję czujnika ultradźwiękowego. By obliczyć odległość przy pomiarze, fala dźwiękowa musi się odbić od mierzonego obiektu. W przypadku tego projektu wykrywane mają być głównie listy, z którymi czujnik ultradźwiękowy może mieć problem, ze względu na właściwości papieru do przewodzenia fali dźwiękowej. By potwierdzić tę tezę szukałem porównania obu czujników, aż znalazłem artykuł „Performance comparison of Infrared and Ultrasonic sensors for obstacles of different materials in vehicle/ robot navigation applications” autorstwa S Adarsh, S Mohamed Kaleemuddin, Dinesh Bose i K I Ramachandran, oparty na licencji postępowania IOP.

W podanym artykule sprawdzona jest różnica odległości mierzonej przy użyciu obu czujników od odległości

rzeczywistej dla różnych materiałów. Z wyników pomiarów dla papieru sporządzona została tabela:

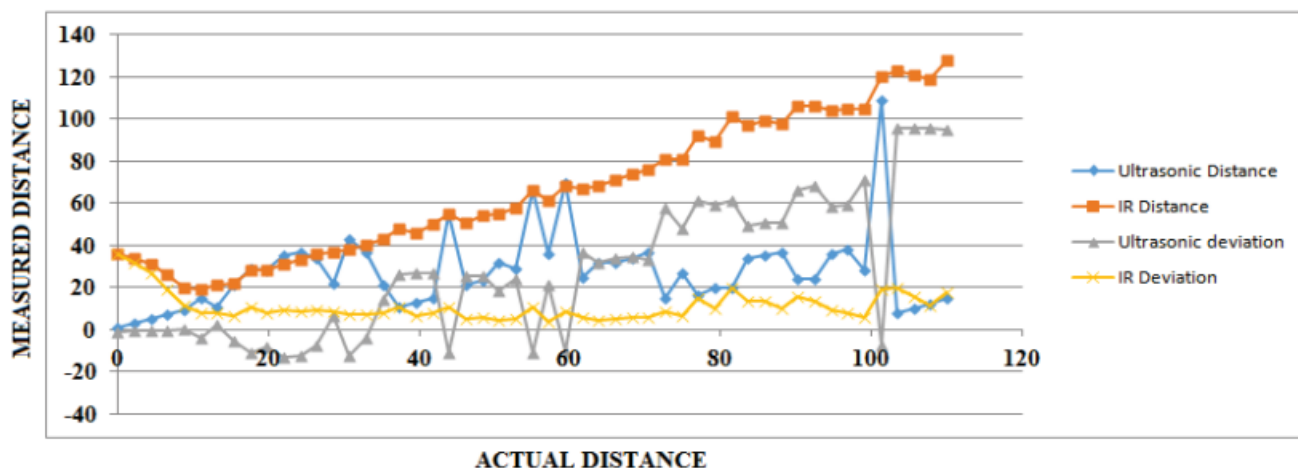


Tabela 2. Wyniki pomiarów czujnika ultradźwiękowego i podczerwieni

Autorzy żółtą linią oznaczyli różnicę dla czujnika podczerwieni, a szarą dla czujnika ultradźwiękowego. Jak widać znacznie lepiej w tym eksperymencie wypada czujnik podczerwieni. Najbardziej interesującą częścią wykresu powinien być początek, ponieważ w projekcie będzie potrzeba mierzenia małych odległości, do około 10 cm. Jak widać na wykresie na samym początku (do 10 cm) badany czujnik daje bardzo nieprecyzyjne wyniki, ale jest to spowodowane specyfikacją czujnika, którego zasięg wynosi od 10 do 80 cm. Natomiast mimo że czujnik ultradźwiękowy daje precyzyjne wyniki do ok. 15cm, to dalej ma bardzo duże różnice i jeśli mam wybierać między tymi dwoma to wolę wykorzystać czujnik podczerwieni, ponieważ nie wiem jak zachowa się, znacznie mniej przewidywalny, czujnik ultradźwiękowy. Dodatkowym problemem są warunki, w których czujnik będzie pracował. Eksperyment z artykułu został przeprowadzony na otwartej przestrzeni, a wykonanie podobnych pomiarów w metalowej skrzynce, w której ma pracować urządzenie może dać zupełnie inne wyniki, na niekorzyść czujnika ultradźwiękowego.

Biorąc pod uwagę wszystkie opisane wyżej problemy, których nie da się rozwiązać w sensowny sposób, w projekcie wykorzystam czujnik podczerwieni.

2.2 Wybór modułu z czujnikiem IR

Zacząłem szukać czujnika odległości IR z wyjściem cyfrowym (I2C), ponieważ są bardziej odporne na zakłócenia i łatwiejsze w obsłudze. Zdecydowałem że wykorzystam czujnik w technologii Time of Flight, który pozwoli na pomiar odległości ściany skrzynki pocztowej od urządzenia i po kalibracji urządzenie będzie co jakiś czas sprawdzać aktualny dystans od najbliższego obiektu. Kierowałem się odległością pomiaru większą od 10 cm, opiniami, temperaturą pracy oraz ceną i po wstępnej analizie wielu urządzeń wybrałem 4 modele:

producent	sensor bazowy	koszt [zł]	odl. min. [mm]	odl. max. [mm]	powierzchnia [cm ²]
DFRobot	VL53L0X	75	30	2000	4,40
Adafruit	VL6180X	90	5	200	3,69
Waveshare	VL53L0X	50	30	2000	4,80
Pololu	VL53L0X	80	0 (???)	2000	2,34

Tabela 3. Porównanie modułów z czujnikami odległości

Patrząc na porównanie można zobaczyć że trzy z czterech modułów wykorzystują ten sam sensor, a mimo to producent Pololu zakłada że jego moduł jest w stanie mierzyć odległość od skrajnie małego dystansu, do czego podchodzę trochę sceptycznie. Po dokładnym przeglądaniu dokumentacji doszedłem do wniosku że wybrane moduły są do siebie bardzo podobne, w końcu na tych samych lub podobnych sensorach, więc kierując się ceną wybrałem moduł producenta Waveshare. W porównaniu do niektórych z reszty modułów brakuje mu wyprowadzeń do złącza goldpin ale nie jest to problem nie do rozwiązania. Posiada stabilizator napięcia 2,5 do 5,5V, a sam działa na 3,3 V. Minimalna odległość pomiaru to 3 cm, więc w fazie testowania produktu zobaczę jakie

wyniki odczytuje sensor z obiektem oddalonym o $< 3\text{cm}$, bo najprawdopodobniej będą to wartości mniejsze od 3cm i nie wpłynie to na poprawne działanie urządzenia (jedynie wprowadzi wymóg pozostawienia tych 3cm przed urządzeniem zamontowanym w pustej skrzynce pocztowej). Moduł czujnika ma pin odpowiedzialny za sterowanie urządzeniem by nie pobierało prądu w czasie gdy pomiar nie jest konieczny.

Zakładam że nie ma potrzeby wykorzystania więcej niż jednego takiego czujnika w urządzeniu, ponieważ powinien bez problemu trafić wiązką na list w skrzynce, jeśli urządzenie będzie zamontowane w odpowiednim miejscu. Jeśli jednak w fazie testowania wystąpią takie problemy, to prawdopodobnie będzie je można rozwiązać w inny sposób niż przez dodanie kolejnego czujnika, co znacznie zwiększyłoby pobór prądu.

2.3 Wybór mikrokontrolera i modułu

Przed określeniem ilości portów GPIO potrzebnych w zadaniu, trzeba określić jakie elementy będą potrzebne, by obsługiwać urządzenie w jak najprostszy sposób. Zdecydowałem że najlepszym rozwiązaniem sterowania urządzeniem będą 2 elementy: suwak odpowiedzialny za włączenie/wyłączenie urządzenia oraz przycisk do kalibracji dystansu urządzenia od ściany skrzynki.

Moduł mikrokontrolera musi mieć przynajmniej następujące wejścia/wyjścia:

Czujnik IR: SDA (I2C) + SCL (I2C) + SHUT (GPIO)

Suwak on/off: 1 input (GPIO)

Przycisk do kalibracji: 1 input (GPIO)

Mierzenie napięcia na baterii: 1 analog input (GPIO)

Łącznie daje to piny obsługujące I2C i 4 piny GPIO, w tym jeden analogowy. Piny cyfrowe powinny działać na $3,3\text{V}$.

Dodatkowo moduł mikrokontrolera musi obsługiwać łączność Wi-Fi.

Do wyboru mikrokontrolera podszedłem niezbyt obiektywnie, ponieważ słyszałem wiele dobrych opinii o ESP32-C3, który spełnia wszystkie wymagania zadania, m. in. wystarczająca liczba portów, porty analogowe, obsługa Wi-Fi, niska cena, napięcie 3,3V i wysoka energooszczędność. Mimo to zapoznałem się z innymi mikrokontrolerami typu ATmega (serie 328), STM32 (serie L, F, WL) czy ESP32 (serie S2, S3, C3, C6). ESP32-C3 okazał się najlepszy do tego zadania, innym mikrokontrolerom brakowało modułu Wi-Fi, były większych rozmiarów, niedostępne na rynku lub drogie. Na drugim miejscu znalazł się ESP32-S3 ale był droższy, większy, miał dużo więcej GPIO niż potrzebne oraz nie zależy mi tak bardzo na szybkości procesora.

Z wybranym mikrokontrolerem zacząłem przeglądać jego moduły na polskim rynku. Zacząłem analogicznie jak w przypadku czujnika odległości, tym razem kierując się m. in. rozmiarem modułu, ceną oraz dostępnością i wybrane modele zawarłem w tabeli:

producent	model	koszt [zł]	powierzchnia [cm2]	zasilanie bateryjne	antena
DFRobot	Beetle	46	5,125	tak	wewnętrzna
Adafruit	QT Py	62	3,916	tak	wewnętrzna
M5Stack	M5Stamp C3U	46	6,8	nie	wewnętrzna
Seeedstudio	Xiao	35	3,675	tak	zewnętrzna

Tabela 4. Porównanie modułów mikrokontrolera ESP32-C3

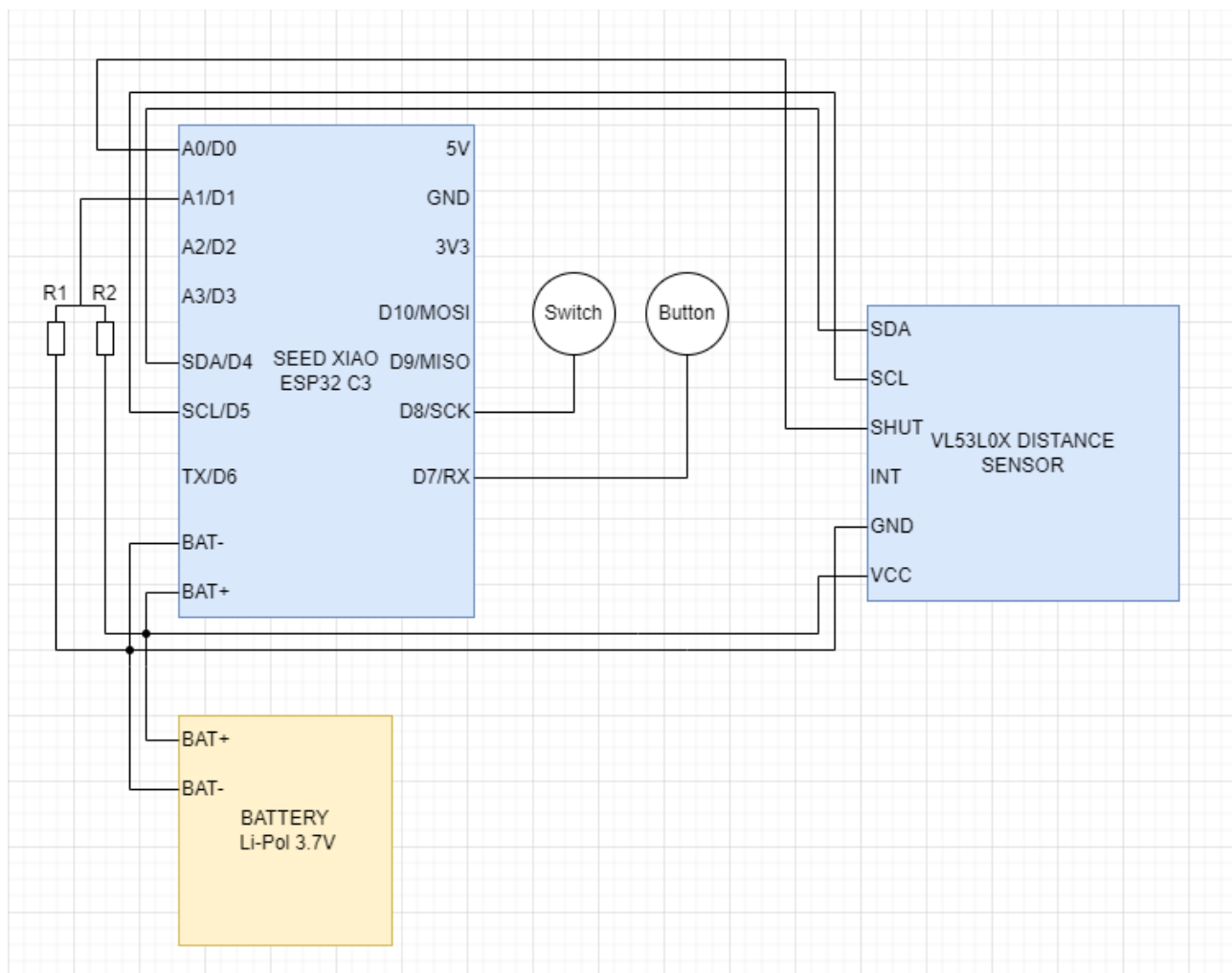
Zdecydowałem że użyję modułu Seeedstudio Xiao ESP32-C3, ze względu na bardzo mały rozmiar, niską cenę, dostępność, popularność oraz możliwość podłączenia akumulatora. Moduł posiada małą zewnętrzną antenę co może wydłużyć zasięg Wi-Fi. Spełnia wszystkie potrzeby tego projektu.

2.4 Zasilanie urządzenia

Czujnik odległości potrzebuje według producenta napięcia 3,3 lub 5V (stabilizator w module przyjmuje 2,5 - 5,5V), a producent modułu mikrokontrolera zaleca wykorzystanie akumulatora 3,7V. Natężenie prądu czujnika w stanie pracy to ~20mA, a modułu Xiao w stanie aktywnym z Wi-Fi to 75mA, a w stanie pół-snu z Wi-Fi to 25mA.

By zasilić urządzenie wykorzystam akumulator Li-Pol 2200mAh z maksymalnym prądem ładowania oraz rozładowania 2,2A, oraz z prądem znamionowym 440mA. Nie powinien mieć żadnych problemów z dostarczeniem wymaganego natężenia prądu, a jego napięcie mieści się w widełkach stabilizatorów czujnika odległości oraz modułu Xiao. Prąd ładowania akumulatora przez moduł wynosi ~380mA co też nie stanowi problemu.

2.5 Schemat blokowy



Rysunek 1. Schemat blokowy

Na powyższym rysunku widać moduł ESP32 z następującymi połączeniami:

D0 – SHUT (IR Sensor)

SDA – SDA (IR Sensor)

SCL – SCL (IR Sensor)

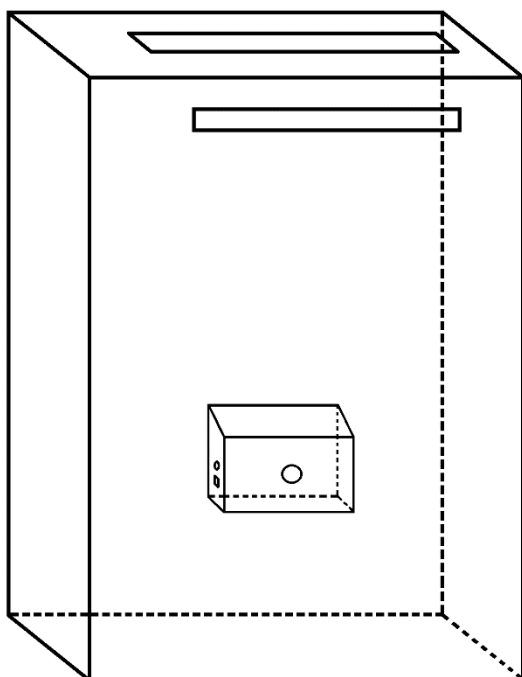
A1 – mierzenie napięcia na akumulatorze

D8 – przełącznik włącz/wyłącz

D7 – przycisk do kalibracji

Dodatkowo na schemacie można zobaczyć źródło zasilania, czyli baterię Li-Pol 3,7V.

2.6 Wizualizacja



Rysunek 2. Wizualizacja urządzenia

Na zewnątrz po boku urządzenie powinno mieć wyprowadzony przycisk i przełącznik do sterowania. W dolnej części obudowy znajdzie się gniazdo do ładowania akumulatora. Górna część obudowy będzie w kształcie trójkąta by uniemożliwić spadanie listów na urządzenie. W przedniej części obudowy znajdzie się szkiełko wyprowadzające wiązkę podczerwieni. Urządzenie dzięki swojej małej masie będzie mogło być montowane na taśmie dwustronnej. Urządzenie będzie montowane w dolnej części skrzynki.