

KIERUNEK: Automatyka i Robotyka (AIR)

PRACA DYPLOMOWA
INŻYNIERSKA

Tytuł pracy:
System zarządzania w zastosowaniach
akwarystycznych

AUTOR:
Bartosz Wroński

PROMOTOR:
dr inż. Wojciech Domski

Niniejszą pracę dedykuję moim rodzicom - Ewie i Andrzejowi oraz przyjaciołom, na których zawsze mogłem liczyć na każdym etapie mojego życia.

STRESZCZENIE

Praca prezentuje sterownik przeznaczony do zarządzania zbiornikiem akwarystycznym. Urządzenie reguluje pracę grzałki, filtra oraz oświetlenia. System został wyposażony w czujniki niezbędne do sterowania urządzeniami akwarystycznymi. Do budowy systemu wykorzystano mikrokontroler oraz mikrokomputer. Mikrokontroler odpowiedzialny jest za pozyskiwanie danych z czujników oraz sterowanie urządzeniami akwarystycznymi. Mikrokomputer został wyposażony w wyświetlacz dotykowy, za pomocą którego można zarządzać zbiornikiem akwarystycznym.

SUMMARY

The thesis presents a driver that allows to control an aquarium tank. The device regulates the work of a heater, a filter and a lightning. The system is equipped with sensors which are necessary to control the aquarium equipment. The system is based on a microcontroller and a microcomputer. The microcontroller is responsible for acquiring data from sensors and controlling the aquarium equipment. The microcomputer is equipped with a touchscreen, which allows to manage the aquarium tank.

Słowa kluczowe: akwarium, sterownik, mikrokontroler, mikrokomputer, Qt, interfejs użytkownika, STM, Raspberry Pi, czujnik, sterowanie

Keywords: aquarium, driver, microcontroller, microcomputer, Qt, user interface, STM, Raspberry Pi, sensor, control

Spis treści

1	Wstęp	3
1.1	Teza	5
2	Obwód elektroniczny	7
2.1	Zasilanie układu	7
2.2	Mikrokontroler STM32	9
2.3	Mikrokomputer Raspberry Pi	10
2.4	Czujniki	11
2.5	Elementy wykonawcze	13
3	Środowisko Qt na mikrokomputerze Raspberry Pi	15
3.1	Projekt interfejsu panelu sterującego	15
3.2	Dostępne funkcjonalności	17
4	Oprogramowanie mikrokontrolera	21
4.1	Środowisko programistyczne	21
4.2	Wykorzystane interfejsy komunikacyjne	23
5	Pomiary zużycia energii elektrycznej	29
6	Podsumowanie	33
	Załącznik A	35
	Załącznik B	36
	Bibilografia	37

Rozdział 1

Wstęp

Świat zwierząt już od dawna fascynował człowieka. Niejednokrotnie ludzkość czerpała inspiracje z natury. Jest ona obecna w wielu dziedzinach: literaturze, muzyce, czy malarstwie. Analogie są także dostrzegalne w świecie technologii. Konstruktorzy maszyn wiele razy dostrzegali unikalne umiejętności występujące u zwierząt, co starano się później odtworzyć. Jako przykład może posłużyć myśliwiec F-14 Tomcat o zmiennej geometrii skrzydeł. Przypominają one w swoim działaniu skrzydła ptaków drapieżnych, które zmieniają swoją geometrię w zależności od prędkości. Innym przykładem może być japoński pociąg Bullet Train, którego kształt inspirowany był zimorodkiem. W dalszym ciągu ludzkość odkrywa niezwykle mechanizmy i zależności występujące w ekosystemach. Do jednej z tych dziedzin należy akwarystyka.

Historia akwarystyki rozpoczęła się w VI wieku n.e. w Chinach. Tamtejsza ludność postanowiła wpuścić do stawów ogrodowych ryby, które miały pełnić ozdobną funkcję. Tym gatunkiem był karaś złocisty, z którego otrzymano później rybę nazywaną potocznie "złotą rybką". Przez kolejne 1000 lat ludzkość hodowała ryby jedynie w stawach. Wiek XVI jest okresem, kiedy można powiedzieć o pierwszej formie akwarium nieco zbliżoną do dzisiejszej postaci. Leonard Baldner, z zawodu rybak, hodował ryby w słojach, których dno przykrywał piasek. Następny krok miał miejsce w 1774 roku, kiedy to zoolog Abraham Trembley postanowił stworzyć szklany zbiornik w kształcie prostopadłościanu. Nie odniósł on znaczących efektów, ze względu na brak warunków przypominających naturalne. W 1830 roku Charles des Moulins udoskonalił metodę hodowli ryb w zbiornikach akwarystycznych wprowadzając natlenianie wody, co znacznie poprawiło warunki bytowe organizmów przebywających w zbiorniku [2].

Wraz z odkryciem elektryczności nastąpiło wiele przełomów w znacznej części dziedzin m. in. również w akwarystyce. Możliwe stało się stworzenie urządzeń utrzymujących odpowiednie warunki w akwarium. Do podstawowych urządzeń tego typu zalicza się filtr. Jest to urządzenie odpowiedzialne za natlenianie oraz oczyszczanie wody. Większość filtrów oferuje jedynie filtrację mechaniczną, która polega na wychwytywaniu zanieczyszczeń fizycznych. Zalicza się do nich fragmenty roślin, odchody ryb lub niezjedzony pokarmu. Elementem gromadzącym zanieczyszczenia jest najczęściej gąbka lub wkład wykonany ze specjalnego materiału takiego jak wata lub włóknina. Lepiej wyposażone urządzenia filtrują wodę biologicznie oraz chemicznie. Filtracja chemiczna polega na zastosowaniu wkładów wykonanych z węgla aktywnego, żywicy lub spieków z dodatkami, które odpowiedzialne są za oczyszczanie wody z substancji pozostałych po lekach, metali ciężkich, fosforanów, czy azotanów. Ostatni rodzaj filtracji, jakim jest filtracja biologiczna, polega na rozkładaniu związków takich, jak amoniak (NH_3/NH_4), związki zawierające jon azotynowy NO_2 oraz związki zawierające jon azotanowy NO_3 przez bakterie nitryfikacyjne

(tlenowe), a następnie przez beztlenowe [6].

Bardzo ważnym aspektem w akwarystyce jest utrzymanie odpowiedniej temperatury wody. Budowa i działanie grzałek jest stosunkowo prosta. Wyróżniamy ich dwa rodzaje. Pierwszy z nich to urządzenia niezawierające termostatu, które działają w sposób ciągły. Istotną wadą tego rozwiązania jest brak możliwości utrzymania temperatury na stałym poziomie. Taka grzałka pracuje w trybie ciągłym, bez względu na temperaturę. Drugi rodzaj to grzałki wyposażone w termostat. Użytkownik, dzięki pokrętle jest w stanie ustawić pożądaną temperaturę w akwarium, którą sterownik grzałki będzie automatycznie regulować.

W akwarium bardzo ważne jest oświetlenie, który jest niezwykle istotny ze względu na vegetację roślin. Do poprawnego rozwoju potrzebują one głównie światła czerwonego o długości fali w przedziale 600-700 nm oraz niebieskiego, którego długość fali waha się w przedziale 400-480 nm. Ważnym czynnikiem jest także temperatura barwowa oświetlenia. Najczęściej spotykaną wartością w oświetleniu akwarystycznym jest 6500K (światło dzienne). Można stosować oświetlenie o niższej temperaturze, wtedy światło wpada w cieplejsze odcienie lub wyższej, barwa staje się chłodniejsza i otrzymane światło ma niebieskie odcienie, a powyżej 10000K barwa staje się fioletoworóżowa. Jeszcze kilkanaście lat temu standardem były świetlówki (lampy fluorescencyjne), które w wyniku wyładowań pomiędzy elektrodami i przy pomocy luminoforu, przekształcając promieniowanie ultrafioletowe, emitowały światło widzialne. To rozwiązanie miało dwie istotne wady. Pierwszą jest konieczność zastosowania statecznika oraz startera. Drugą wadą jest spadek jasności wraz z czasem. W akwarystyce świetlówki należało wymieniać raz do roku, ze względu na duże zużycie lampy. Od kilku lat w oświetleniu dominują rozwiązanie bazujące na technologii LED. LEDy przewyższają lampy fluorescencyjne pod kątem żywotności oraz efektywności energetycznej. Ważnym parametrem w przypadku lamp LED jest współczynniki oddawania barw CRI lub Ra. Parametr Ra podawany jest w zakresie od 0 do 100, gdzie wartość 100 odpowiada światłu słonecznemu, którego widmo jest ciągłe. Dla zbiornika akwarystycznego zalecana wartość wynosi $Ra > 90$. Ważnym parametrem jest również natężenie oświetlenia. Dla oświetlenia typu LED przyjęto następujący przelicznik. Roślinom o małym zapotrzebowaniu wystarczy oświetlenie o natężeniu 20 lm/l, a dla najbardziej wymagających przyjęto 40lm/L [12].

Ostatnim aspektem, który został poruszony w pracy dotyczy parametrów chemicznych wody. Do najważniejszych zalicza się: poziom Ph, natlenienie, stężenie CO_2 , węglany, twardość GH i KH. Do innych zalicza się stężenie: azotynów, azotanów, fosforanów, chloru oraz żelaza. Wartości niektórych substancji zależą od jakości wody kranowej. Można je kontrolować za pomocą specjalnych testów, a do unormowania służą odpowiednie preparaty.

Ze względu na wyżej przytoczone aspekty, celem niniejszej pracy inżynierskiej było stworzenie układu sterowania, który będzie automatyzował działanie urządzeń elektrycznych wykorzystywanych w akwarium. Do tych urządzeń zalicza się grzałkę, filtr oraz oświetlenie. Zbudowany system wyposażony jest w czujniki. Pierwszy, sensor BH1750, był odpowiedzialny za pomiar natężenia oświetlenia, aby w przypadku pojawienia się źródła światła innego, niż to w akwarium, mikrokontroler dostosował moc oświetlenia. Algorytm sterowania zostanie oparty na regulatorze PID. Jest to typ regulatora, który sprawdza się w większości zastosowań (około 90% przypadków), gdzie wymagane jest sterowanie konkretną wartością. Ostatnią funkcjonalnością, która zostanie zaimplementowana, będzie odpowiedzialna za stopniowe włączanie i wyłączenie oświetlenia tak, aby cały cykl przypominał wschód i zachód słońca. Kolejny aspekt dotyczy pracy grzałki, która będzie działała w sprzężeniu zwrotnym z czujnikiem DS18B20. Działanie będzie oparte o pętlę

histerezy. Temperatura zostanie utrzymana w konkretnym przedziale, uwzględniając bieżącą godzinę. Jest to spowodowane zmieniającą się ceną za energię elektryczną w trakcie doby. Cały układ wyposażony będzie w czujnik HC-SR04, którego zadaniem jest pomiar poziomu cieczy. Jest to w istocie zabezpieczenie, które ma zadziałać w przypadku wycieku ze zbiornika. Wszystkie zmierzone wartości zaprezentowane będą na wyświetlaczu dotykowym, za którego działanie będzie odpowiedzialny mikrokomputer Raspberry Pi. Dzięki takiemu rozwiązaniu możliwe będzie monitorowanie wszystkich parametrów oraz zmiana zadanych wartości. Dodatkowo zaimplementowane zostaną dwa tryby pracy dla oświetlenia oraz grzałki. Pierwszy to tryb automatyczny, który został opisany powyżej. Natomiast drugi ręczny, nie będzie wykorzystywał informacji z czujników. Wszystkie wartości będą zadawane przez użytkownika.

1.1 Teza

Celem niniejszej pracy inżynierskiej jest stworzenie układu sterowania przeznaczonego do wykorzystania w akwarystyce przy pomocy mikrokontrolera współpracującego z mikrokomputerem.

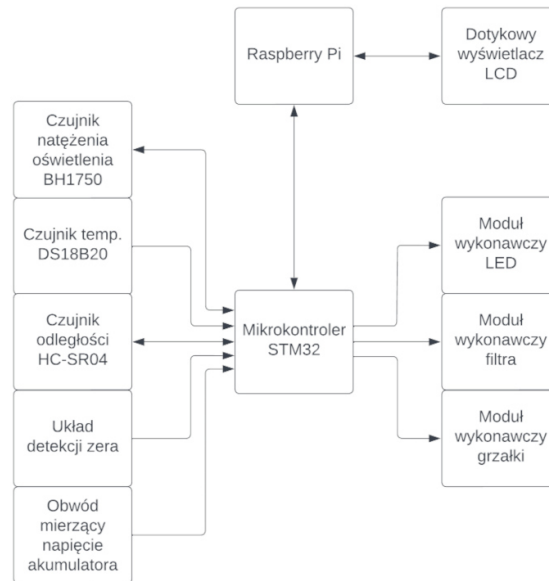
Rozdział 2

Obwód elektroniczny

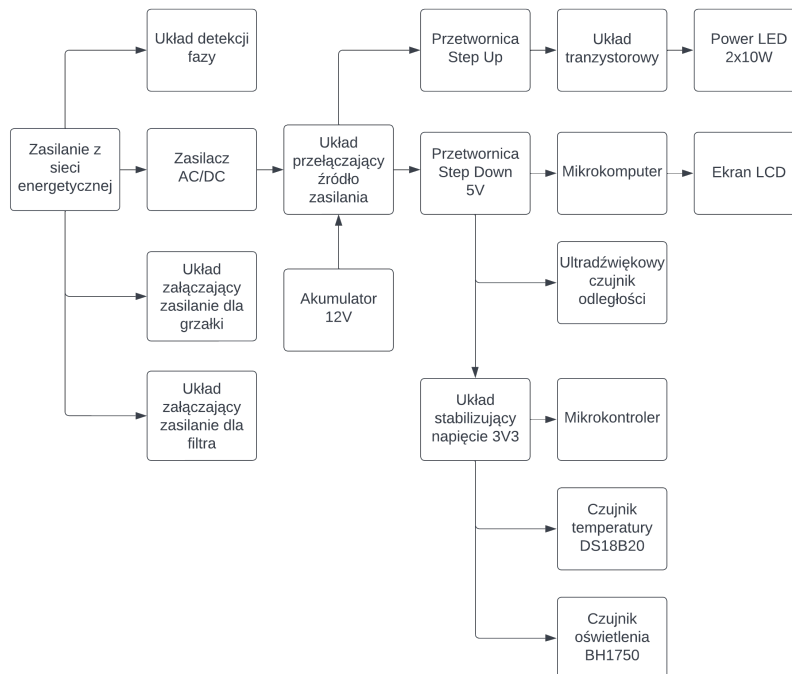
Do poprawnego działania większości współczesnych urządzeń, oprócz oprogramowania niezbędne są obwody elektroniczne oraz układy scalone. Pełnią one rozmaite funkcje w zależności od zastosowania, które obejmują wszystkie dziedziny współczesnego świata. Obraz dzisiejszej cywilizacji po części można zawdzięczać rewolucji krzemowej, która nastąpiła w połowie ubiegłego wieku. John Bardeen oraz Walter Houser Brattain opracowali wtedy pierwszy działający typ tranzystora. Jednak bez skutecznej metody produkcji rewolucja krzemowa nie nastąpiłaby w tak szybkim tempie. Sukces na tym polu należy do Polaka, Jana Czochralskiego, który stworzył metodę uzyskiwania monokryształu krzemu w drugiej dekadzie XX wieku, tym samym wyprzedzając swoje czasy. Stworzenie niniejszego sterownika było możliwe dzięki wcześniej wymienionym wynalazkom. Głównymi zadaniami, jakie miał realizować wcześniej wspomniany system, było stworzenie układu zasilania, zapewniającego energię elektryczną dla mikrokomputera, wyświetlacza LCD, mikrokontrolera oraz oświetlenia. Następnym zadaniem do zrealizowania było zaprojektowanie układu służącego do detekcji momentu, w którym napięcie na przewodzie fazowym wynosi 0 V. Miało to na celu umożliwienie regulacji mocy grzałki, odpowiedzialnej za podgrzewanie wody. Cała metoda regulacji mocy, opiera się na czasie po jakim należy załączyć układ z triakiem, co umożliwi przepływ prądu elektrycznego przez odbiornik. Ostatni obwód, który należało wykonać miał za zadanie umożliwić sterowanie urządzeniami, za pomocą mikrokontrolera, pracującymi pod napięciem 230 V prądu przemiennego. Wszystkie elementy, z których składał się system zostały przedstawione na rys. 2.1, w celu lepszego zobrazowania idei działania całego urządzenia.

2.1 Zasilanie układu

Sekcję zasilania sterownika, przedstawioną na rys. 2.2 można podzielić na dwie części. Pierwszą z nich jest część zasilana, gdzie energia była zapewniona bezpośrednio z sieci energetycznej, dostarczającej napięcie przemiennie o wartości 230 V. Do układów pracujących, pod wcześniej wymienionym napięciem, należał układ detekcji fazy oraz dwa układy wykonawcze oparte na triakach. Były one odpowiedzialne za włączanie urządzeń zasilanych napięciem przemiennym 230 V. Z kolei do drugiej części urządzeń i układów można zaliczyć urządzenia zasilane napięciem stałym znacznie niższym, niż w przypadku wcześniej wymienionego źródła. W niniejszym systemie kluczem było osiągnięcie trzech różnych napięć, do których należały 3,3 V, 5 V oraz 30 V. Napięcie 3,3 V zostało wykorzystane do zasilania mikrokontrolera, cyfrowego czujnika temperatury DS18B20 oraz czujnika natężenia oświetlenia BH1750. Napięcie uzyskano przy pomocy liniowego stabilizatora



Rysunek 2.1 Architektura systemu



Rysunek 2.2 Topologia zasilania

LV33CV [11]. Należy on do rodziny układów o oznaczeniu LFXX, która charakteryzuje się niskim napięciem odcięcia (z ang. *low-dropout voltage*) 0,45 V, przy którym stabilizator jest w stanie poprawnie pracować. Kolejną zaletą jest bardzo niski prąd spoczynkowy (z ang. *quiescent current*), który w trybie OFF wynosi 50 μV , a w trybie ON 500 μV . Do wad tego rozwiązania należy sprawność regulatorów liniowych, która waha się w przedziale 42-78 % [8]. Pomimo niskiej sprawności w porównaniu z stabilizatorami impulsowymi, zapewniają one dużo krótszy czas reakcji na zmianę napięcia wejściowego, które wynosi 5-50 μs . Wszystkie wcześniej wymienione cechy, popularność oraz cena sprawiły, że zdecydowano się na takie rozwiązanie. Kolejne napięcie, czyli o wartości 5 V, zostało uzyskane za pomocą przetwornicy obniżającej (z ang. *step-down converter*). Urządzenia tego typu przetwarzają napięcie stałe, na napięcie o mniejszej wartości, które można regulować najczęściej za pomocą potencjometru obrotowego. Przetwornice dostępne na rynku przeważnie można zakupić w postaci niewielkiej płytki PCB. Pomimo większych rozmiarów w porównaniu ze stabilizatorami liniowymi czy impulsowymi, zdecydowano się na to rozwiązanie. Elementem, który przeważał, była konieczność, aby moduł był w stanie dostarczyć 3 A, a więc tyle, co rekomendowany zasilacz do mikrokomputera Raspberry Pi4. W przypadku stabilizatorów liniowych pojawiłyby się straty w postaci wydzielanego ciepła, które w przybliżeniu równe są iloczynowi różnicy napięcia pierwotnego oraz wtórnego i prądowi płynącemu przez stabilizator. Podczas testów, w trakcie których zmieniano źródło zasilania, z zasilacza dostarczającego 20 VDC na akumulator żelowy o napięciu 12 VDC, przetwornica utrzymała napięcie w zakresie tolerancji. Dzięki temu sterownik jest w stanie działać przez pewien czas na własnym źródle zasilania. Napięcie na akumulatorze jest mierzone przez mikrokontroler, dzięki czemu w przypadku zbyt niskiego poziomu naładowania zostanie to zasygnalizowane na wyświetlaczu. Dodatkowo zmierzono zapotrzebowanie systemu czujników, mikrokontrolera oraz mikrokomputera wraz z wyświetlaczem na energię, która wyniosła 6,06 W oraz 4,53 W przy wyłączonym wyświetlaczu.

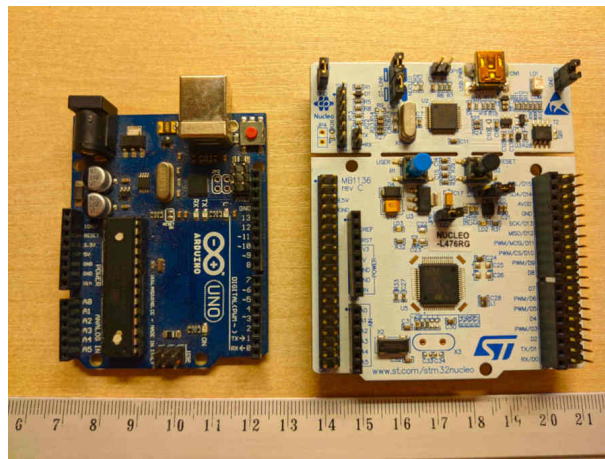
2.2 Mikrokontroler STM32

Podczas wyboru platformy należało określić niezbędne wymagania. Na tamtą chwilę istniało kilka platform, które mogły zrealizować wymagane zadania. Do tych platform można zaliczyć Arduino, które cechuje się stosunkowo niskim progiem wejścia. Najpopularniejsze modele, czyli Arduino Uno, które widoczne jest na rys. 2.3 oraz Mini wyposażone są w mikrokontroler Atmega 328. Dodatkowo do dyspozycji użytkownika dostępne są takie peryferia, jak SPI (z ang. *Serial Peripheral Interface*), I^2C (z ang. *Inter-Integrated Circuit*), cyfrowe i analogowe wejścia, cyfrowe wyjścia oraz port szeregowy. Kolejnym atutem tej platformy jest dedykowane IDE, które cechuje niezwykle prosty interfejs, ale przez to niemożliwa jest zaawansowana konfiguracja urządzenia. Istnieje również możliwość zainstalowania dodatku Platform IO do programu Visual Studio Code. Innym rozwiązaniem brany pod uwagę było ESP32. Jest to dedykowana platforma do rozwiązań IoT (z ang. *Internet of Things*) znacznie przewyższająca możliwości platformy Arduino ze względu na liczbę dostępnych interfejsów, do których należą: SPI (z ang. *Serial Peripheral Interface*), I^2C (z ang. *Inter-Integrated Circuit*), I^2S (z ang. *Inter-IC Sound*), SDIO (z ang. *Secure Digital Input Output*), UART (z ang. *Universal async receiver transmitter*), CAN (z ang. *Controller Area Network*), ETH (z ang. *Ethernet MAC*) oraz IR (z ang. *Infrared*). Dodatkowo ESP32 ma wbudowany czujnik temperatury (zakres od -40°C do $+125^\circ\text{C}$), przetwornik cyfrowo-analogowy DAC (Digital-to-analog converter), analogowo-

cyfrowy z przybliżeniem sukcesywnym SAR ADC (ang. *Successive approx analog-to-digital converter*) oraz PWM (z ang. *pulse width modulation*). Trzecim urządzeniem rozważanym podczas wyboru było Raspberry Pi Pico. Wyposażone jest ono na bardzo podobnym poziomie, co wcześniej wymienione ESP32, jednak nie zawiera modułów łączności. Czwartym i ostatnim urządzeniem wziętym pod uwagę była platforma STM Nucleo. Jest ona wyposażona w wiele peryferiów i interfejsów komunikacyjnych, co pozwala na obsługę wielu zewnętrznych urządzeń. Do programowania przeznaczone jest dedykowane IDE – STM32CubeMX. Istnieje jeszcze możliwość programowania mikrokontrolerów Stm za pomocą Platform IO. Wszystkie wcześniej wymienione platformy programowane są w językach C/C++, w przypadku Raspberry Pi Pico, ESP32 oraz STM Nucleo można wybrać MicroPythona. Odnosząc się do Arduino należy zaznaczyć, że MicroPython będzie wprowadzony oficjalnie, jako część tego środowiska. Na chwilę obecną MicroPython jest w fazie testowej. Po rozważeniu wszystkich zalet i wad każdego rozwiązania stwierdzono, że najlepszym wyborem będzie płytki deweloperska STM Nucleo. Wyposażona jest w procesor ARM (z ang. *Advanced RISC Machine*) Cortex-M4, o maksymalnej częstotliwości pracy wynoszącej 80 MHz. Dostępnych jest wiele interfejsów, takich jak I^2C , SPI, UART oraz CAN. Zestaw wyposażony jest we wbudowany programator oraz debugger ST-Link/V2-1, dzięki czemu do programowania wymagany jest jedynie przewód mini-USB typu B. Możliwość debuggowania okazała się niezwykle istotna w trakcie realizacji pracy inżynierskiej, ponieważ pomogło to zlokalizować newralgiczne miejsca w oprogramowaniu. Do zadań realizowanych przez mikrokontroler należało sterowanie filtrem, odpowiedzialnym za filtrację mechaniczną, biologiczną i chemiczną, które odbywało się w sposób binarny. Na pinie w zależności od użytkownika był wystawiany stan wysoki (3,3 V), bądź niski (0 V). Z kolei do zarządzania pracą grzałki oraz oświetlenia wykorzystano sygnał o zmiennym stopniu wypełnienia. Jest to sygnał, który znany jest pod nazwą PWM (z ang. *pulse width modulation*), dzięki czemu możliwe jest wygenerowanie napięć pośrednich w zakresie 0-3,3 V. Umożliwiło to płynną regulację mocy grzałki oraz ilości światła emitowanego przez diody typu power LED, odpowiadające za doświetlenie zbiornika akwarystycznego. Należało także nawiązać komunikację z czujnikiem temperatury DS18B20, aby utrzymać odpowiednią temperaturę w zbiorniku akwarystycznym, która najczęściej waha się w przedziale 20-28 °C. Wpływ na to ma szerokość geograficzna oraz biotop z jakiego pochodzą organizmy, które znajdują się w akwarium. Kolejnym czujnikiem, z którym należało się skomunikować służył do regulacji mocy oświetlenia. Konieczna była także obsługa czujnika ultradźwiękowego służącego do pomiaru odległości od tafli wody, a więc ilości wody znajdującej się w zbiorniku akwarystycznym. Do ostatnich elementów realizowanych przez STM Nucleo należała obsługa przerw generowanych przez układ detekcji zera oraz mierzenie napięcia na awaryjnym źródle zasilania, jakim był akumulator żelowy. Wprowadzono to rozwiązanie w celu zabezpieczenia go przed osiągnięciem zbyt niskiego napięcia, a tym samym nieodwracalnego uszkodzenia. W przypadku powyższego akumulatora, którego znamionowe napięcie wynosiło 12 V, napięcie krytyczne było równe 10,5 V. Ta wartość oznaczała, że akumulator był całkowicie rozładowany i należy go podłączyć do ładowarki, aby dalsze korzystanie stało się możliwe.

2.3 Mikrokomputer Raspberry Pi

W celu realizacji założeń projektowych, do których należało stworzenie graficznego interfejsu użytkownika, w skrócie GUI (z ang. *graphical user interface*), należało wybrać inną platformę niż STM Nucleo. Na rynku istnieją dedykowane wyświetlacze do mikro-



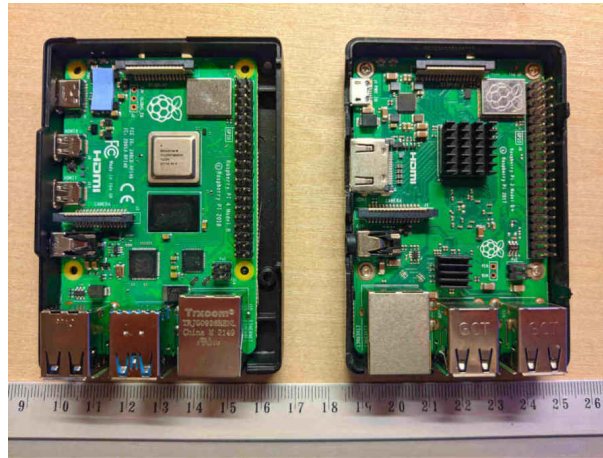
Rysunek 2.3 Platforma Arduino Uno po lewej, po prawej STM Nucleo L476RG

kontrolerów STM32, choćby do serii Discovery. Jednak próg wejścia w przypadku tego typu urządzeń jest stosunkowo wysoki. Bardzo dobrym rozwiązaniem do realizacji tego typu zadań okazał się mikrokomputer. Dysponują one dużo większą mocą obliczeniową niż mikrokontrolery, porównywalną nawet z komputerami sprzed około 20 lat. Zachowują przy tym rozmiary nieznacznie większe od typowej karty płatniczej. Zapotrzebowanie na energię wynosi maksymalnie kilkanaście watów, co jest dodatkową zaletą. Na chwilę obecną na rynku jest dostępnych wiele urządzeń tego typu. Do najpopularniejszych należą Raspberry Pi, Intel Edison, Odroid, czy FriendlyARM. Najpopularniejszą platformą bez wątplenia jest Raspberry Pi, co było decydujące podczas wyboru. Zdecydowano się na model 4B [1], który wyposażony jest w 4 GB pamięci RAM (z ang. *Random Memory Access*). Istnieją również wersje z 2 lub nawet 8 GB pamięci operacyjnej, jednak taka ilość nie była wymagana. Ze względu na zastosowanie innego typu procesorów niż w standardowych laptopach czy komputerach stacjonarnych konieczna jest instalacja specjalnie przeznaczonych systemów operacyjnych do tego typu urządzeń. Ze względu na dużą popularność Raspberry Pi powstało wiele systemów dedykowanych pod konkretne zastosowania. Do wyboru są systemy przeznaczone typowo do rozrywki, emulacji gier, systemy stworzone do druku 3D oraz kilka wersji systemów bazujących na Linuxie. Na Raspberry Pi 4, widocznym na rys. 2.4 w wersji wyposażonej w 4 lub 8 GB pamięci RAM możliwa jest instalacja Ubuntu. Największą popularnością cieszy się dystrybucja Linuxa stworzona typowo na mikrokomputery Raspberry Pi, uznana za system podstawowy, czyli Raspbian lub często nazywany Raspberry Pi OS. Jest to dystrybucja oparta na Debianie i rozwijana jako wolne oprogramowanie na zasadach licencji GNU General Public License. W przypadku niniejszego projektu wybór padł na wcześniej wymieniony system Raspberry Pi OS w wersji 64 bit. Umożliwiło to uruchomienie środowiska Qt, co zostało dokładniej opisane w rozdziale 3.

2.4 Czujniki

W projekcie niezbędne było zastosowanie czujników odpowiedzialnych za akwizycję danych z otoczenia. W realizowanym projekcie sensory były odpowiedzialne za pozyskiwanie informacji na temat temperatury wody znajdującej się w zbiorniku akwarystycznym, natężenia oświetlenia oraz ilości wody znajdującej się w akwarium.

Pierwszy z sensorów, którego zadaniem był pomiar temperatury w zbiorniku musiał



Rysunek 2.4 Od lewej, Raspberry Pi model 4B oraz Raspberry Pi model 3B+



Rysunek 2.5 Czujnik DS18B20 w wariacie wodoodpornym [10]

być wyposażony także w hermetyczną obudowę pozwalającą na bezpieczne zanurzenie w cieczy. Urządzeniem, które spełniło powyższe wymagania był cyfrowy czujnik temperatury DS18B20, widoczny na rys. 2.5, który pozwala na pomiar temperatury w zakresie od -55 do $+125$ °C. W projekcie zakres temperatur przeważnie waha się w przedziale 20 – 28 °C, w zależności od biotopu znajdującego się w akwarium. Dokładność pomiaru wynosi $0,5$ °C, co jest wystarczające i ewentualny błąd pomiarowy nie powoduje jakiegokolwiek ryzyka związanego z egzystencją organizmów w zbiorniku.

Do poprawnego działania czujnika zalecane są trzy przewody: zasilanie 3.3 V, masa GND oraz przewód sygnałowy. Istnieje również możliwość podłączenia dwu-przewodowego, w przypadku którego wyeliminowany jest przewód zasilający. W takiej sytuacji energia elektryczna jest dostarczana za pomocą przewodu sygnałowego i ma miejsce wtedy tzw. “zasilanie pasożytnicze”. Takie funkcjonowanie sensora umożliwia interfejs 1-Wire, który został opisany w rozdziale 4.2.

Drugi czujnik jest odpowiedzialny za pomiar odległości od tafli wody. Jest to bardzo popularny moduł HC-SR04, pokazany na rys. 2.6, który swoje działanie opiera na ultradźwiękach. Ultradźwięki posiadają tę własność, że bardzo dobrze sprawdzają się w przypadku pomiarów odległości od płaszczyzn. Zakres pomiarowy czujnika mieści się w przedziale 2 – 450 cm. W projekcie był mierzony bardzo mały zakres odległości, który sięgał od około 8 do 20 cm. Pomiar odbywa się w następujący sposób. Po podaniu wysokiego stanu logicznego o długości 10 μ s na linię Trigger, czujnik emituje dźwięk o częstotliwości 40 kHz i jednocześnie wystawia wysoki stan logiczny na linii Echo. W momencie odbioru fali



Rysunek 2.6 Czujnik ultradźwiękowy HC-SR04 [5]



Rysunek 2.7 Czujnik natężenia światła BH1750 [4]

dźwiękowej przez mikrofon zostaje wystawiony niski stan logiczny na linii Echo. Dzięki pomiarowi długości trwania stanu wysokiego na linii Echo można obliczyć odległość czujnika od obiektu. Do podłączenia modułu wykorzystano cztery przewody: zasilający 5 V, masa GND, Trigger oraz Echo. Trzeci i ostatni czujnik służy do pomiaru natężenia oświetlenia. Sensor BH1750 swoją budowę opiera o fototranzystor, którego wartość prądu emitera zależy od liczby padających fotonów na bazę. Umożliwia to, za pomocą kilku innych komponentów, na mierzenie wartości w przedziale od 1 do 65535 lx. W przypadku sterownika do akwarium mierzone wartości osiągały minimalnie 0 lx w nocy i przy wyłączonym oświetleniu, do maksymalnie 4000 lx przy maksymalnej mocy diod Power LED. Sensor zasilany jest napięciem o wartości 5 V, a komunikacja odbywa się przy pomocy interfejsu I^2C , do którego działania wymagane są dwie linie - danych oraz zegarowa.

2.5 Elementy wykonawcze

Bardzo ważną częścią sterownika są układy, które umożliwiły sterowanie urządzeniami wykonawczymi. Do tego typu układów zalicza się układ sterujący oświetleniem, filtrem oraz grzałką. Do zasilania urządzeń wykorzystano dwa rodzaje napięć o różnych wartościach. Do zasilania diod power LED wykorzystano stałe napięcie o wartości 30 V. Z kolei grzałka oraz filtr były zasilane bezpośrednio z sieci elektrycznej napięciem przemiennym o wartości 230 V.

W przypadku diod Power LED, zapewniających odpowiednie warunki do przeprowadzania fotosyntezy przez rośliny, układ sterujący oparty jest o tranzystor bipolarny typu NPN (BD139). Idea działania układu bazuje na modulacji szerokości impulsu, który generowany jest przez mikrokontroler. Pozwala to na kluczowanie tranzystora, a więc włączanie i wyłączanie go z odpowiednią częstotliwością. Zmiana stosunku czasu, kiedy tranzystor znajduje się w stanie przewodzenia do czasu zatkania umożliwia regulację ilości światła emitowanego przez diody. Częstotliwość pracy klucza w sterowniku wyniosła 100 Hz dzięki czemu efekt migotania diod był niezauważalny przez ludzkie oko.

Urządzenia pracujące na wcześniej wspomnianym zasilaniu z sieci wymagały układu, który pozwoli na ich bezpieczne włączanie za pomocą mikrokontrolera. Do tego celu stworzono układ, który składał się z optotriaka MOC3021 wyzwalającego bramkę triaka BT137. Dzięki zastosowaniu optotriaka, w układzie występowała separacja galwaniczna,

przez co mikrokontroler był zabezpieczony przed uszkodzeniem przez napięcie sieciowe. Niezbędnym elementem do regulacji ilości ciepła wydzielanego przez grzałkę był układ wykrywający moment w którym napięcie fazowe zmieniało znak na przeciwny. Układ oparty był na transoptorze PC817, co zapewniło separację galwaniczną, a więc zabezpieczenie w przypadku awarii po stronie pierwotnej układu. W momencie zmiany znaku napięcia układ generował zbocze opadające, które było wykrywane przez mikrokontroler, jako przerwanie. W następnej kolejności odmierzano czas w przedziale 0-10 ms, po którym następowało załączenie triaka BT137. Im krótszy czas, tym wydzielana moc była bliższa maksymalnej.

Rozdział 3

Środowisko Qt na mikrokomputerze Raspberry Pi

Środowisko Qt należy do frameworków ułatwiających tworzenie aplikacji z interfejsem graficznym. Qt współpracuje z wieloma platformami, dzięki czemu aplikacje mogą być przenoszone na inne urządzenie w stosunkowo łatwy sposób. Środowisko przeznaczone jest do programowania w wielu językach: C++, Java oraz QML. Istnieje też możliwość programowania w języku Python, jednak wymaga to większej wprawy. Qt stosuje mechanizm sygnałów i slotów, który służy do przesyłania zdarzeń w aplikacjach graficznych. Sygnał emitowany jest w przypadku wystąpienia zdarzenia, czyli np. naciśnięcie przycisku, zmiana pozycji suwaka czy wprowadzenie danych do pola tekstowego. Slot zajmuje się obsługą zdarzenia, które wystąpiło i tam należy umieścić wszystkie metody oraz funkcje, które mają zostać wykonane.

3.1 Projekt interfejsu panelu sterującego

Realizacja projektu od samego początku zakładała wykorzystanie wyświetlacza dotykowego. Taka decyzja była spowodowana wcześniejszymi doświadczeniami podczas tworzenia zegara opartego na lampach NIXIE LC-513. Do sterowania były wykorzystywane cztery przyciski typu tact switch, które po około roku przestały zwierać obwód, a tym samym spełniać swoją funkcję. Inny projekt, który był realizowany i dostarczył cennego doświadczenia wykorzystywał klawiaturę membranową, która również uległa zużyciu po około 18 miesiącach. Istotną zaletą jest łatwość modyfikacji całego programu i wprowadzanie poprawek lub dodawanie nowych elementów. Ze względu na brak ruchomych elementów dotykowe wyświetlacze cechują się bardzo długą żywotnością. Chociaż warto podkreślić, że istnieją technologie, które do detekcji punktu dotyku wykorzystują zewnętrzną warstwę wyświetlacza, która pod wpływem nacisku dotyka inną. Zamykany jest wtedy obwód elektryczny, który zbudowany jest z niewidzialnych elektrod, o określonej rezystancji, wkomponowanych w tafle wyświetlacza. W różnych miejscach rezystancja ulega zmianie i umożliwia to na obliczenie współrzędnych. Ten typ wyświetlaczy określa się mianem ekranów rezystancyjnych. Istnieją również ekrany wykorzystujące technologię opartą o fale akustyczne lub podczerwień. Na obecną chwilę jednak największą popularnością cieszą się pojemnościowe ekrany dotykowe, które umożliwiają obsługę funkcji multi-touch. Polega ona na wykrywaniu kilku punktów dotyku jednocześnie, z czego w największej mierze korzystają urządzenia takie, jak smartfony obsługując różne gesty. Zastosowanym wyświetlaczem w niniejszym projekcie jest ekran pojemnościowy o przekątnej 7", który

wykonany jest w technologii IPS (z ang. *In-Plane Switching*). Jest to technologia, która została zaprezentowana przez firmę Hitachi w 1996 roku, jako odpowiedź na matryce TN (z ang. *Twisted Nematic*), które były tanie w produkcji, cechowały się krótkim czasem reakcji, jednak pokrycie barw oraz kąty widzenia były na bardzo niskim poziomie. Wyświetlacze IPS rozwiązały część problemów takich, jak wyżej wymienione małe kąty widzenia oraz częściowe pokrycie barw. Jednak pogorszeniu uległy czasy reakcji matrycy, występuje efekt smużenia oraz słabsze odwzorowanie czerni w porównaniu z innymi barwami. Często można spotkać również matryce typu VA (z ang. *Vertical Alignment*), które pod względem parametrów można zakwalifikować pomiędzy ekranami korzystającymi z technologii TN oraz IPS. Cechują się słabszym odwzorowaniem barw, jednak zachowują dobre kąty widzenia, niewiele gorszy czas reakcji zapewniając przy tym znacznie lepszy kontrast oraz czerń. W przypadku matryc IPS ciemne barwy mogą przypominać odcienie szarości. Różnica parametrów tych matryc wynika z ułożenia ciekłych kryształów względem powierzchni panelu. W przypadku matryc TN kryształy znajdują się pomiędzy szklanymi płytkami i ich pozycja wyjściowa jest prostopadła do panelu. W przypadku matryc VA oraz IPS kryształy ułożone są kolejno wielokierunkowo oraz równoległe do płaszczyzny ekranu. Obecne na rynku są również wyświetlacze typu AMOLED (z ang. *Active Matrix Organic Light-Emitting Diode*), które w swojej budowie korzystają z organicznych diod niewymagających dodatkowego podświetlenia. W przypadku czerni niesie to ze sobą bardzo istotną zaletę. Ze względu na to, że dioda samodzielnie emituje światło, może być również całkowicie wyłączona, co sprawia, że czerń jest wiernie oddawana. Z kolei wadą tego rozwiązania jest ryzyko związane z nierównomiernym zużywaniem się poszczególnych diod, a więc wypalaniem się ekranu. Do takiej sytuacji może doprowadzić wyświetlanie statycznych obrazów przez długi czas. Spowodowało to, że ten typ wyświetlacza nie nadaje się do niniejszego projektu. W aplikacji zmianie będą ulegać jedynie niewielkie obszary, na których prezentowane są poszczególne pomiary. Doprowadziło to, że finalnie zastosowano wyświetlacz IPS. Innym istotnym parametrem podczas wyboru wyświetlacza był jego rozmiar oraz rozdzielczość. W przypadku wybranego wyświetlacza zdecydowano się na matrycę o rozdzielczości 1024x600 i przekątnej 7", aby osiągnąć wystarczającą ostrość czcionek i zapewnić ilość miejsca na wszystkie elementy interfejsu. Rozmiar elementów dobrany został w sposób empiryczny, tak jak i kolorystyka. Zdecydowano się na ciemne barwy, aby w przypadku obsługi przy braku innego oświetlenia użytkownik nie odczuwał dyskomfortu. Sam interfejs, którego finalna wersja znajduje się na rys. 3.2 składa się ze stosunkowo wielu pól i na pozór wydaje się dosyć skomplikowany, dla porównania, tak prezentował się wstępny projekt interfejsu, który pokazano na rys. 3.1. Można wyróżnić część interfejsu odpowiedzialną za pracę grzałki, gdzie ukazana jest moc, jaka jest oddawana do wody w celu utrzymania odpowiedniej temperatury. Pokazywany jest obecny stan grzałki, która może być włączona lub wyłączona. Kolejnym elementem jest obecna temperatura, która mierzona jest za pomocą czujnika DS18B20. Nieco niżej znajduje się pole tekstowe, gdzie prezentowana jest temperatura, do której ma dążyć sterownik. Kolejną sekcją jest oświetlenie. Wyświetlane są tutaj wartości takie jak: zadane natężenie oświetlenia podawane w luksach [lx], analogicznie, jak w przypadku grzałki, nieco niżej znajduje się aktualne natężenie oświetlenia mierzone za pomocą czujnika BH1750. W centralnej części interfejsu można dostrzec pola związane z czasem. Pokazana jest planowana godzina włączenia i wyłączenia. Aktualna godzina wyświetlana jest w dolnym lewym rogu w polu "Informacje komunikacyjne". Ostatnie dwa pola prezentują tryb oświetlenia (automatyczny/ręczny) oraz stan (włączone/wyłączone). Przedostatnią częścią interfejsu jest obszar, gdzie pokazywane są dane związane z zasilaniem. Pole "Źródło zasilania" mówi użytkownikowi o tym, czy sterownik zasilany jest bezpośrednio z sieci elektroenergetycznej



Rysunek 3.1 Wstępny projekt interfejsu



Rysunek 3.2 Finalny projekt interfejsu

lub z drugiego źródła awaryjnego, którym jest akumulator żelowy. Dwa ostatnie elementy w tej części mówią o napięciu, które mierzone jest w bezpośredni sposób za pomocą dzielnika napięcia. Z kolei pasek statusowy dostarcza informacji na temat poziomu naładowania akumulatora w graficzny sposób. Ostatnim polem jest terminal, który wyświetla komunikaty dotyczące statusu komunikacji pomiędzy mikrokontrolerem STM oraz mikrokomputerem Raspberry Pi. W przypadku nawiązywania połączenia można odczytać, jakie czynności wykonuje aplikacja, a następnie komunikaty w przypadku ewentualnych problemów. Jeśli komunikacja przebiega pomyślnie, w oknie wyświetlana jest jedynie aktualna godzina.

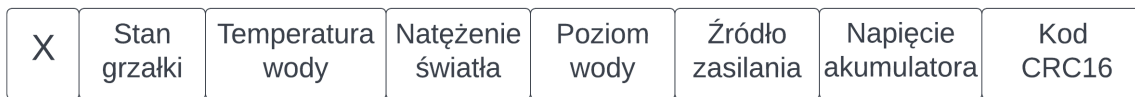
3.2 Dostępne funkcjonalności

Bardzo istotnymi elementami w trakcie projektowania interfejsów graficznych oprócz aspektu wizualnego są funkcjonalności, które umożliwiają interakcję oraz sterowanie wartościami nastaw, parametrów czy stanem urządzeń. Niniejszy sterownik także jest wyposażony w elementy, z którymi użytkownik może wchodzić w interakcje. Do dyspozycji są cztery suwaki determinujące moc grzałki, zadaną temperaturę, natężenie (tylko kie-

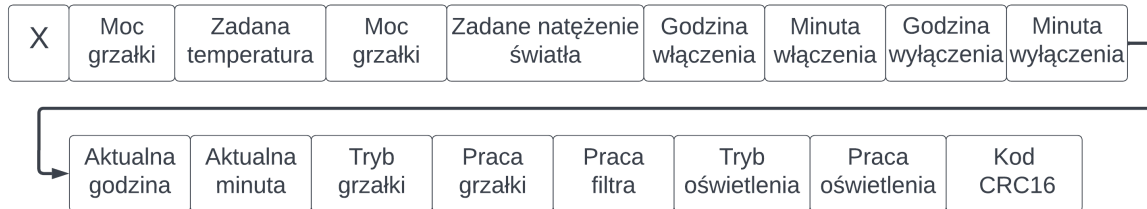
dy oświetlenie jest w trybie automatycznym) oraz moc oświetlenia (tylko podczas pracy oświetlenia w trybie ręcznym). W centralnej części GUI (z ang. *Graphical User Interface*) znajdują się także przyciski, które pozwalają na przełączanie się pomiędzy poszczególnymi trybami urządzeń. Dla filtra, odpowiedzialnego za oczyszczanie i natlenianie wody, oświetlenia, zapewniającego warunki do przeprowadzania fotosyntezy i grzałki dostarczającej ciepła dostępna jest możliwość uruchamiania i wyłączenia, przy pomocy przycisków “Włącz” i “Wyłącz”. Dla oświetlenia i grzałki jest dostępny dodatkowo tryb ręczny oraz automatyczny, wyzwalany za pomocą przycisków “Auto” i “Manual”. Użytkownik posiada także możliwość ustawiania godziny włączenia i wyłączenia oświetlenia przy pomocy specjalnych pól z przyciskami. Na samym początku po uruchomieniu aplikacji należy nawiązać komunikację z mikrokontrolerem. W tym celu należy najechać na górne pole o nazwie “Połączenie”, a następnie na opcję “Szukaj”. W tej chwili w terminalu powinny pojawić się stosowne komunikaty związane z wyszukiwaniem urządzeń. Jeśli cały proces przebiegł pomyślnie, po najechaniu na opcję “Urządzenia” powinna pokazać się lista dostępnych urządzeń. Po wybraniu mikrokontrolera STM powinna rozpocząć się dwukierunkowa komunikacja pomiędzy systemami. Program został stworzony w taki sposób, aby za komunikację był odpowiedzialny osobny wątek. Sama idea wątków polega na tym, że program wykonywany jest współbieżnie. Każdy wątek ma dostęp do tych samych zasobów pamięci w obrębie danego procesu, co pozwala na szybsze wykonywanie poszczególnych zadań. W przypadku tej aplikacji jest to odbiór i przetwarzanie informacji, które docierają z mikrokontrolera. Ramka danych została zaprojektowana w taki sposób, aby wychwytywać niepoprawne wartości. W tym celu na początku został umieszczony identyfikator pod postacią znaku “X”, dodatkowo na końcu przychodzącej ramki znajduje się kod CRC (z ang. *Cyclic Redundancy Check*, CRC), czyli cykliczny kod nadmiarowy. Sprawdza się on w przypadku eliminacji losowych błędów, dlatego jest szeroko stosowany w komunikacji przewodowej, jak i bezprzewodowej. Sam algorytm polega na dzieleniu ciągu znaków (danych) przez n-bitowy wielomian, czego wynikiem jest reszta z dzielenia. W praktyce najczęściej korzysta się z CRC16 lub CRC32. Istnieje znaczna liczba odmian wielomianów, które są stosowane w zależności od obszaru zastosowania. Dla kodu CRC16 istnieją: CRC-16-Chakravarty, CRC-16-ARINC, CRC-16-CCITT, CRC-16-CDMA2000, CRC-16-DECT, CRC-16-T10-DIF, CRC-16-DNP, CRC-16-IBM, CRC-16-OpenSafety-A, CRC-16-OpenSafety-B oraz CRC-16-Profibus. Biorąc pod uwagę charakterystykę niniejszego urządzenia zdecydowano się na zastosowanie CRC-16-IBM, który jest określany bardzo często jako CRC-16, bez żadnego przedrostka. Wielomian ten opisany jest jako liczba heksadecymalna 0x8005, co przekłada się na wielomian o następującej postaci

$$x^{16} + x^{15} + x^2 + 1.$$

Wielomian ten stosowany jest w przypadku transmisji danych za pomocą szyny USB (z ang. *Universal Serial Bus*). Istotną wadą tego rozwiązania niewątpliwie jest brak zabezpieczenia przed sytuacją, kiedy dane zostały celowo sfalszowane, dlatego to zabezpieczenie nie sprawdzi się w przypadku kryptografii. Należy też pamiętać, że kombinacja dwóch lub więcej błędów w wyniku dzielenia może dać identyczny wynik, jak w przypadku poprawnych danych, chociaż jest to bardzo mało prawdopodobna sytuacja. Po odebraniu ramki i weryfikacji poprawności danych następuje parsowanie danych, czyli wyodrębnienie z ciągu znaków poszczególnych wartości takich, jak temperatura, czy natężenie światła. Cała ramka danych została przedstawiona na rys. 3.3. W analogiczny sposób odbywa się konstruowanie i wysyłanie ramki danych, widoczne na rys. 3.4 do mikrokontrolera, jednak składowych elementów jest zauważalnie więcej. Komunikacja odbywa się w sposób asynchroniczny, czyli wysyłanie danych oraz odbiór odbywa się w sposób niezależny. Dane są



Rysunek 3.3 Odbierana ramka danych przez Raspberry Pi



Rysunek 3.4 Wysyłana ramka danych przez Raspberry Pi

wysyłane cyklicznie dzięki wykorzystaniu układu liczącego z częstotliwością równą 5 Hz. W wyniku tego zdarzenia uruchamiany był slot, który odpowiedzialny był za wysyłanie ramki danych do mikrokontrolera. Wyższa częstotliwość nie była konieczna ze względu na obszar zastosowania sterownika. Prędkość transmisji ustawiona została na 115200 bps, co odpowiadało prędkości transmisji około 14 kB/s. Długość ramki danych wynosiła 51 znaków, gdzie każdy był reprezentowany za pomocą 8 bitów, co przełożyło się na objętość 0,051 kB. Maksymalna częstotliwość wysyłania wcześniej wspomnianej ramki danych przy prędkości komunikacji na poziomie 14 kB/s mogła wynieść maksymalnie 270,51 Hz. Jest to wynik teoretyczny, który brał pod uwagę jedynie przepustowość magistrali i nie uwzględniał jakichkolwiek opóźnień w wykonywaniu programu na mikrokomputerze. Powyższe obliczenia miały na celu pokazanie jak wiele informacji można przesłać w ciągu sekundy za pomocą stosunkowo wolnego połączenia. Warto wziąć pod uwagę możliwości światłowodów, dzięki którym prędkość przesyłania danych w sieci Internet osiąga nie rzadko 1 Gb/s. Przy takiej prędkości wyżej wymienioną ramkę można byłoby wysłać teoretycznie z częstotliwością na poziomie 2,45 MHz. Jest to jedynie jeden z wielu interfejsów, które oferowane są w urządzeniach tego typu. W rozdziale 4.2 opisano interfejsy, z których korzysta mikrokontroler STM w celu komunikacji z czujnikiem DS18B20 odpowiedzialnym za odczyt temperatury wody oraz sensor BH1750, którego zadaniem był pomiar natężenia światła.

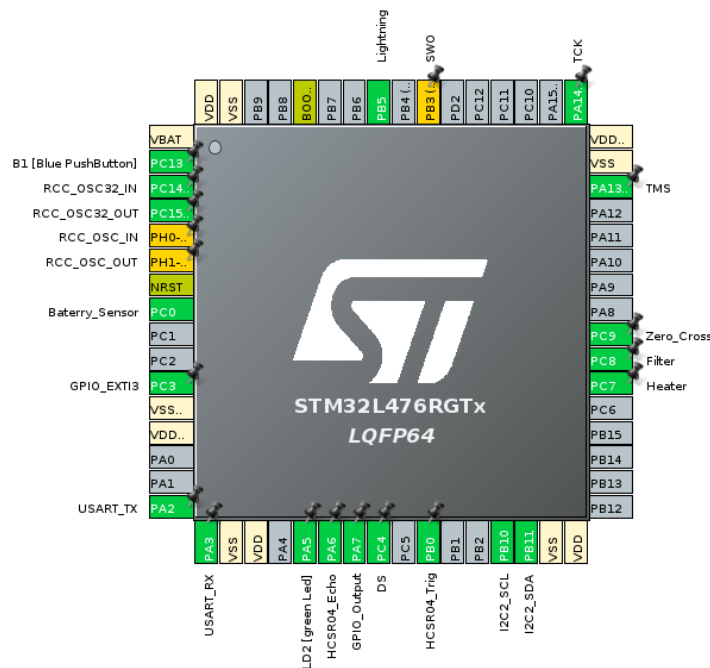
Rozdział 4

Oprogramowanie mikrokontrolera

Kluczowym elementem w trakcie realizacji projektu sterownika akwarystycznego oprócz układów i obwodów elektronicznych było stworzenie odpowiedniego oprogramowania na mikrokontroler. Tak jak było wspomniane we wcześniejszym rozdziale, platformą, która została wybrana jest STM Nucleo L476RG. Jest to płytki deweloperska przeznaczona do tworzenia systemów wbudowanych. Systemy wbudowane są obecne w wielu dziedzinach współczesnego świata. Obszarami zastosowania tego typu układów są urządzenia sterujące pracą silników samochodowych, systemy ABS, oprogramowanie samolotów, raket, bankomatów lub elektrowni. Nawet urządzenia AGD oraz RTV są wyposażone w tego typu systemy. Niniejszy sterownik również można określić jako system wbudowany, ponieważ realizuje ściśle określone zadania, co jest istotą tego typu urządzeń. Mikrokontrolery oraz mikroprocesory pracujące w segmencie *embedded* współpracują z urządzeniami takimi, jak czujniki, czy napędy. W zbudowanym sterowniku tymi elementami są: czujnik odległości HC-SR04, sensor natężenia oświetlenia BH1750, czujnik temperatury DS18B20, układ mierzący napięcie na akumulatorze, oraz trzy układy zapewniające możliwość sterowania oświetleniem, grzałką oraz filtrem.

4.1 Środowisko programistyczne

W przypadku tworzenia oprogramowania dla rozmaitych systemów niezwykle ważne jest środowisko, w którym tworzone jest oprogramowanie. Dobór odpowiedniego narzędzia jest kluczowy z punktu widzenia efektywnej i bezproblemowej pracy. Podczas wyboru narzędzia zdecydowano się na STM32CubeMx ze względu na wcześniejsze doświadczenie i znajomość tego środowiska. STM Cube dostarcza możliwość konfiguracji mikrokontrolera w sposób graficzny, co jest bardzo istotnym ułatwieniem. W mikrokontrolerach ARM występuje duży stopień skomplikowania i rozbudowania timerów oraz interfejsów, które niejednokrotnie występują podwójnie lub potrójnie. Ręczne pisanie plików konfiguracyjnych w przypadku tej rodziny mikrokontrolerów z pewnością doprowadziłaby do znacznego wydłużenia samych prac nad projektem i absorbowałyby zbyt dużo uwagi. Mnogość dostępnych opcji i możliwości konfiguracji może być przytłaczająca dla nowych użytkowników w porównaniu z mikrokontrolerami Atmega. W przypadku niniejszego urządzenia należało skonfigurować kilka peryferiów. Należały do nich timery, odpowiedzialne za realizację funkcji czasowych, zegary poszczególnych interfejsów, interfejsy, priorytety przerwań oraz piny mikrokontrolera. Piny mikrokontrolera pełnią różne funkcje, do których należą: obsługa cyfrowych wejść oraz wyjść, odczyt analogowych wartości, w przypadku odczytu napięcia oraz generowanie sygnału PWM dla obwodu odpowiedzialnego za oświetlenie

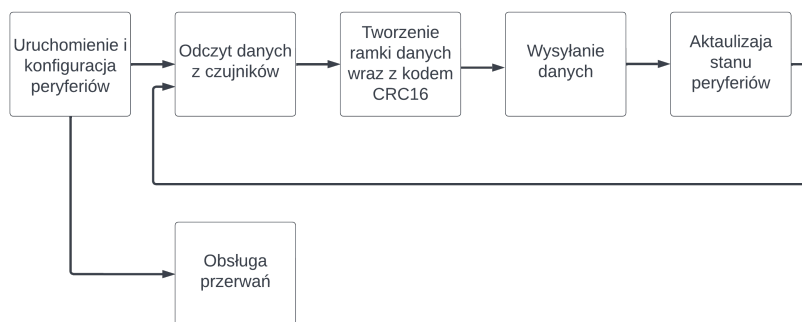


Rysunek 4.1 Konfiguracja pinów

akwarium, tak, jak na rys. 4.1. Po dobraniu odpowiednich parametrów i skonfigurowaniu peryferiów nadszedł czas tworzenia oprogramowania. Na początku zdeterminowano oczekiwania związane z oprogramowaniem, do których należała obsługa ultradźwiękowego czujnika odległości HC-SR04, obsługa czujnika BH1750, sprawdzającego natężenie oświetlenia, sensora DS18B20, odczytującego temperaturę wody w zbiorniku akwarystycznym. Dodatkowo należało odczytywać napięcie na akumulatorze, w odpowiedni sposób sterować oświetleniem, filtrem oraz grzałką. Istotny był również aspekt komunikacji, która musiała być dwukierunkowa, do tego należało również odpowiednio przetwarzać oraz konstruować ramki danych. Wszystkie te elementy sprawiły, że program jest dosyć rozbudowany i składa się z wielu plików, w celu lepszej przejrzystości działania całego programu.

Całe oprogramowanie zostało stworzone w języku C, który jest bardzo często stosowany w systemach wbudowanych. Posiada on szereg zalet, o których warto wspomnieć. Język C udostępnia funkcje sterujące, które determinują kolejność wykonywania danego algorytmu, co znacznie pomaga w wielu sytuacjach. Kolejną zaletą jest możliwość programowania modularnego, strukturalnego oraz zstępującego (z ang. *top-down planning*). Następną zaletą języka C jest jego efektywność [9] wynikająca z wysokiej precyzji, utożsamianej niejednokrotnie z assemblerem, który należy do języków niskiego poziomu. Dzięki temu można zredukować ilość potrzebnego miejsca na kod programu lub zwiększyć prędkość działania. Kolejną ważną cechą języka C jest jego przenośność, czyli możliwość uruchamiania programów na różnych platformach, co czyni go bardzo uniwersalnym. Bardzo ciekawym faktem jest to, że duża część systemu o nazwie UNIX została napisana w języku C, tak samo, jak wiele kompilatorów i interpreterów do języków takich, jak: FORTRAN, Perl, Logo czy Python. Wadą języka C, co niedawno zostało przedstawione, jako zaleta jest możliwość używania wskaźników do zarządzania pamięcią, co może być przyczyną trudnych do wytropienia błędów.

W trakcie pisania kodu wykorzystano w dużej mierze z biblioteki HAL (z ang. *Hardware Abstraction Layer*), która umożliwia dużo łatwiejszą współpracę z peryferiami mikrokon-



Rysunek 4.2 Uproszczona idea działania programu

trolera. Niekiedy można spotkać opinie programistów, że biblioteka HAL zawiera błędy i nie należy jej stosować. Jednak dla programu, który został stworzony w ramach pracy inżynierskiej nie zaobserwowano jakichkolwiek problemów. Działanie programu można opisać w sposób przedstawiony na rysunku 4.2. Ciekawym rozwiązaniem, które zastosowano w przypadku odczytu napięcia z akumulatora, odpowiedzialnego za awaryjne zasilanie, zastosowano rozwiązanie oparte o *DMA* (z ang. *Direct memory access*), które zapewnia bezpośredni dostęp do pamięci, a więc szybsze działanie programu. Moduł DMA działa w następujący sposób. Na początku procesor dostarcza informacji skąd moduł DMA ma odczytywać dane oraz w którym obszarze pamięci RAM mają zostać zapisywane. Po skończonym procesie procesor jest powiadamiany i może się odwołać do odpowiedniego miejsca w pamięci operacyjnej. W przypadku braku tego modułu, procesor jest odpowiedzialny za kopiowanie danych z rejestrów do pamięci operacyjnej, co powodowało dodatkowy narzut czasowy w działaniu programu. Jest to zjawisko szczególnie niekorzystne w przypadku systemów czasu rzeczywistego, które muszą działać możliwie jak najszybciej.

4.2 Wykorzystane interfejsy komunikacyjne

Kluczową rzeczą dla dla całego projektu były interfejsy komunikacyjne, które umożliwiły komunikację pomiędzy mikrokomputerem oraz mikrokontrolerem, a także pomiędzy mikrokontrolerem, a czujnikami BH1750 oraz DS18B20. Wszystkie wymienione połączenia wykorzystują inny protokół komunikacyjny.

Pierwszy z nich, z którego korzysta termometr cyfrowy DS18B20 nazywa się *1-Wire*. Jest to bardzo prosty interfejs, który został stworzony do łączenia urządzeń na znaczne odległości, mogące wynosić nawet kilkaset metrów. Swoją nazwę zawdzięcza faktowi, iż do poprawnego działania urządzenia wymagany jest jedynie jeden przewód, za pomocą którego odbywa się komunikacja dwukierunkowa. Cały moduł lub czujnik można połączyć za pomocą dwóch lub trzech przewodów. W przypadku pierwszej konfiguracji przewodami, które należy podpiąć, są to: masa (GND) oraz przewód sygnałowy, który odpowiedzialny jest za przesył danych oraz zasilanie. W takim przypadku mówi się o zasilaniu pasożytniczym. W przypadku konfiguracji trójprzewodowej wykorzystywane są dwa przewody zasilające oraz jeden przewód sygnałowy. Połączenie może być zrealizowane za pomocą dowolnego przewodu, jednak im lepsza jego jakość, tym tworzone sieci mogą być bardziej rozbudowane oraz rozległe. Najczęściej urządzeniami, które wykorzystują ten interfejs są, termometry cyfrowe, przetworniki ADC, pamięci, ekspandery magistrali 1-Wire oraz układy typu *IO*. Interfejs zakłada, że istnieje jedynie jedno urządzenie nadrzędne (z ang. *master*) oraz dowolna liczba urządzeń podrzędnych (z ang. *slave*), które identyfikowane

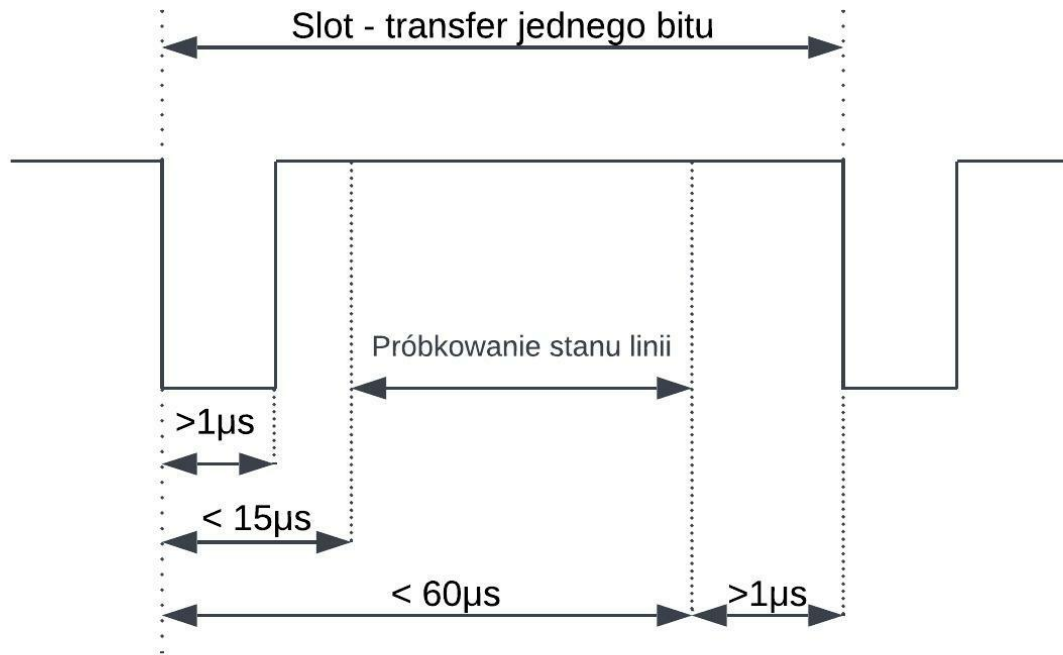
są za pomocą 8-bajтового identyfikatora, który jest przypisywany już w trakcie produkcji modułu. Urządzenia pracujące, jako podrzędne nie wykazują żadnej aktywności, za cały przepływ informacji odpowiedzialny jest jednostka nadrzędna. Wymiana danych, ze względu na jeden przewód, odbywa się szeregowo, począwszy od najmniej znaczącego bitu. Stanem recesywnym magistrali można określić moment, kiedy przewód ma potencjał równy napięciu zasilania. Kosztem, jaki należy ponieść w związku z dużymi odległościami oraz jednym przewodem sygnałowym jest niska prędkość przesyłu danych. Wynosi ona typowo 16 kbps, jednak istnieje możliwość zwiększenia prędkości przesyłu do 125 kbps, lecz wymagany jest do tego specjalny tryb (z ang. *overdrive*). Na rysunku 4.3 przedstawiono schemat komunikacji, który pomaga w zrozumieniu działania tego interfejsu. Za pomocą magistrali 1-Wire można dokonywać operacji odbierania oraz wysyłania danych, które inicjowane są tylko i wyłącznie przez urządzenie nadrzędne. Na samym początku urządzenie nadrzędne wystawia logiczny stan zero, który trwa od 1 do 3 μs . Podczas operacji zapisu następuje przesłanie bitu danych. W momencie wysłania bitu, którego wartość wynosi logiczne zero, magistrala zostaje wprowadzona w stan niski na czas, który wynosi od 15 do 120 μs . Po impulsie inicjalizującym, w momencie przesłania bitu o wartości 1, magistrala wraca do stanu wyjściowego. Po takim cyklu wysłanie kolejnego bitu danych możliwe jest dopiero po 60 μs . Odczyt danych przez urządzenie nadrzędne wygląda w sposób analogiczny tj. po wystawieniu krótkiego impulsu 0 na magistrali, czeka na bit wysłany przez urządzenie podrzędne. W tym celu urządzenie nadrzędne po czasie wynoszącym minimalnie 15 μs próbuje stan magistrali. Jednak każda operacja na magistrali rozpoczyna się od sekwencji sygnałów nazywanych jako sygnał resetujący (z ang. *reset pulse*). Działa ona w następujący sposób. Urządzenie nadrzędne na początku wystawia logiczny stan 0, który trwa na magistrali przez okres od 480 do 960 μs . Po tej sekwencji urządzenie nadrzędne oczekuje na informację zwrotną od urządzeń podrzędnych w postaci sygnałów obecności (z ang. *presence pulse*). Jest to ujemny impuls, który trwa od 60 do 240 μs , zazwyczaj rozpoczyna się w czasie od 15 do 60 μs po sygnale obecności. Impuls ten pozwala określić urządzeniu nadrzędnemu, czy do magistrali podpięte jest jakiegokolwiek urządzenie. Jeśli po wcześniej wymienionym czasie urządzenie master nie otrzyma tego sygnału to znaczy, że komunikacja nie jest możliwa. W przeciwnym przypadku, urządzenia podrzędne rozpoczynają oczekiwanie na polecenie, których lista została przedstawiona na tabeli 4.2. Diagram komunikacji został przedstawiony na rysunku 4.4, gdzie widać, że po inicjalizacji magistrali następuje wysłanie komendy *match rom* lub *skip rom* oraz identyfikator urządzenia, który zajmuje 8 bajtów. W tym momencie urządzenie jest gotowe do komunikacji. Następnie wysyłane jest polecenie, które urządzenie podrzędne ma wykonać oraz opcjonalne parametry dla wybranych komend. Po tym cykl się kończy, w celu realizacji kolejnej komendy należy powtórzyć wszystkie wyżej wymienione czynności.

Następnym interfejsem, o którym należy wspomnieć jest interfejs I^2C , który czasami nazywany jest również TWI (z ang. *two wire serial interface*). W niniejszym projekcie wykorzystywany jest do obsługi czujnika natężenia światła BH1750, niezbędnego do poprawnej regulacji mocy oświetlenia. W porównaniu do interfejsu 1-Wire, I^2C korzysta on z większej liczby przewodów, w których skład wchodzi: masa (GND), dodatni biegun zasilania (Vcc), przewód SCL (z ang. *serial clock*) oraz SDA (z ang. *serial data*), które są liniami dwukierunkowymi. Ze względu na występowanie linii zegarowej, interfejs ten należy do grupy interfejsów synchronicznych. Niewątpliwą wadą tego interfejsu jest brak możliwości przesyłu danych na większe odległości, co oferuje 1-Wire. Wartości te mogą maksymalnie wynosić kilkadziesiąt centymetrów. W celu osiągnięcia komunikacji na większe odległości należy zastosować takie urządzenia jak m.in. wzmacniacze sygnału. Częstotliwość pracy jest kilkakrotnie wyższa od interfejsu 1-Wire, ponieważ I^2C pra-

cuje zazwyczaj w dwóch trybach, chociaż istnieje również trzeci tryb, ale jest rzadziej spotykany. Do tych trybów należą: tryb standardowy, gdzie częstotliwość pracy osiąga wartość równą 100 kHz, w trybie szybkim wartość ta jest czterokrotnie wyższa, wynosi więc 400 kHz oraz rzadko spotykany tryb wysokiej częstotliwości, która wynosi 3,4 MHz, lecz jest obsługiwana przez niewielką liczbę urządzeń [7]. Obie linie, czyli zegarowa oraz linia danych pracują w trybie otwartego drenu (z ang. *open-drain*), który powoduje, że urządzenia podrzędne oraz nadrzędne mogą powodować stan niski, czyli tożsamy z masą (GND). Za stan wysoki odpowiadają rezystory podciągające, dzięki temu możliwa jest praca w trybie wielu urządzeń nadrzędnych (z ang. *multimaster*). Diagram przedstawiający strukturę całej magistrali został przedstawiony na rysunku 4.5. Do magistrali tego typu możliwe jest podłączenie 128 urządzeń. Jest to liczba teoretyczna, ze względu na charakter adresu urządzeń, który składa się z 7 bitów. W części urządzeń pierwsze cztery bity są determinowane przez producenta na etapie produkcji i są wykorzystywane w celu identyfikacji typu urządzenia, którymi mogą być zegar, akcelerometr, czujnik zbliżeniowy itp. Ostatnie 3 bity nazywane są fizycznym adresem urządzenia [7]. Taka kombinacja pozwala na podłączenie jednocześnie 8 urządzeń tego samego typu, co w niektórych przypadkach może okazać się ograniczeniem. Budowę takiego adresu przedstawiono na rysunku 4.6. Cała transmisja zarządzana jest przez urządzenie nadrzędne, pomimo tego, że komunikacja odbywa się dwukierunkowo. Urządzenie nadrzędne generuje sygnał zegarowy, który determinuje prędkość przesyłu danych. Urządzenie podrzędne w przypadku zbyt szybkiej transmisji może ją zwolnić poprzez wystawienie zbyt długiego stanu niskiego na linii *SCK*. Daje to informację zwrotną urządzeniu nadrzédnemu, że układ podrzędny nie jest w stanie nadać lub dane, które mają zostać wysłane nie zostały jeszcze przygotowane. Całą transmisję danych można przedstawić w sposób opisany poniżej:

1. Urządzenie nadrzędne przesyła bit startu.
2. Następuje przesłanie adresu konkretnego urządzenia podrzédnego.
3. Określany jest najmłodszy bit bajtu adresowego, który określa rodzaj operacji: 1 - odczyt, 0 - zapis.
4. Urządzenie podrzédne przesyła bit *ack*, który potwierdza odbiór danych i gotowość do pracy.
5. Następuje wysterowanie linii zegarowej przez urządzenie nadrzędne, w taki sposób, aby nastąpiło nadanie lub odebranie 8 bitów z informacjami.
6. W zależności, czy wcześniej wystąpiła operacja wysłania lub odbioru, zostaje wysłany, bądź odebrany bit *ack*.
7. Na koniec urządzenie nadrzędne wysyła sygnał informujący o zakończeniu cyklu.

Warto zaznaczyć, że kroki o numerach 5 oraz 6 mogą być powtarzane wielokrotnie, w przypadku większej ilości danych. Bit *ack* (z ang. *acknowledge*), który był wcześniej wspomniany posiada wartość równą 0. W przypadku braku tego bitu, a raczej jego poprawnej wartości, urządzenie nadrzędne powinno zresetować stan magistrali, poprzez zainicjalizowanie nowej transmisji od bitu *start*. Należy też wspomnieć o drugiej możliwości, jaką jest wysłanie bitu *nack* (z ang. *negative acknowledge*). Taka sytuacja ma najczęściej miejsce, w momencie, kiedy urządzenie nadrzędne otrzyma więcej bajtów danych, niż jest to spodziewane, wówczas urządzenie nadrzędne nadaje *nack*, który ma wartość



Rysunek 4.3 Schemat komunikacji na magistrali 1-Wire w funkcji czasu

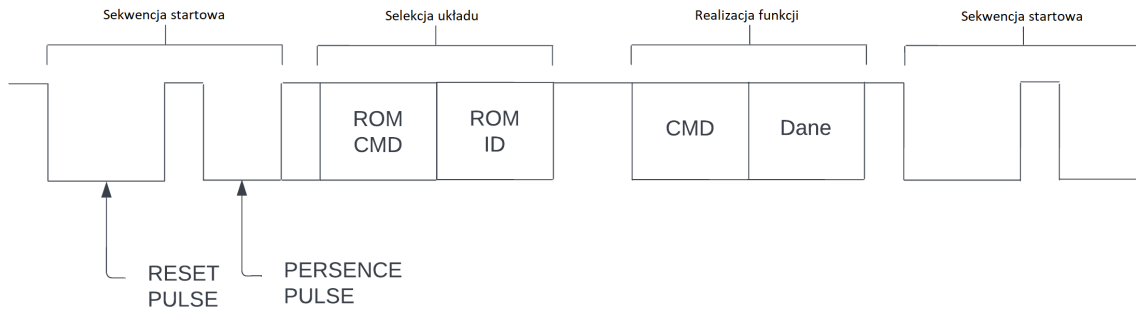
równą 1, a w następnej kolejności emitowany jest sygnał *stop*, który powoduje zakończenie transmisji. Trzeba pamiętać, że nadanie sygnału *nack* zmienia stan linii *scl*. Wcześniej wspomniano również o rezystorach podciągających, które są konieczne do poprawnego działania całego interfejsu. Ich wartość musi być dobrana wedle wzoru ukazanego poniżej. Na rysunku 4.7 przedstawiono zależność wartości rezystorów od pojemności przewodów magistrali I^2C . Minimalna wartość rezystancji określana jest jako

$$R_{min} = \frac{V_{cc} - 0,4V}{3mA},$$

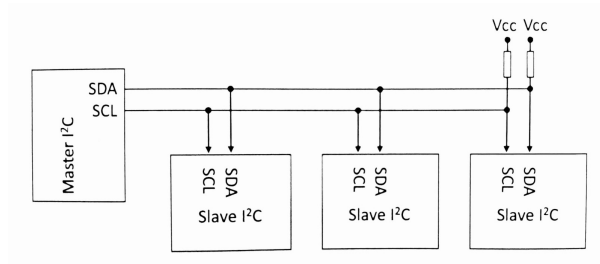
gdzie:

- R_{min} rezystancja minimalna wyrażona w $k\Omega$
- V_{cc} wartość napięcia zasilania podana w V

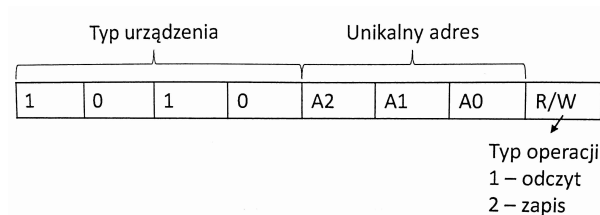
Wartość maksymalna uzależniona jest od częstotliwości pracy magistrali, dlatego dla częstotliwości < 100 kHz wynosi ona 1000 ns/C magistrali, a dla częstotliwości > 100 kHz ma wartość 300 ns/C.



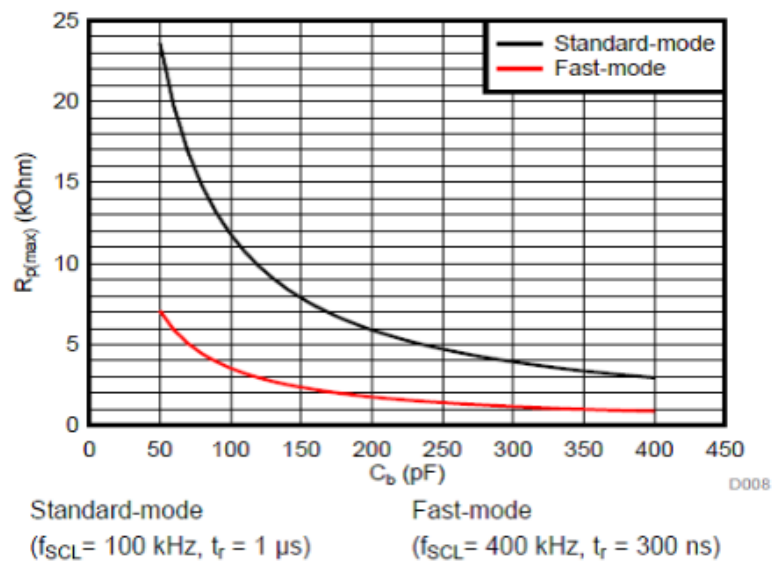
Rysunek 4.4 Schemat komunikacji na magistrali 1-Wire [7]



Rysunek 4.5 Schemat magistrali I²C [7]



Rysunek 4.6 Schemat magistrali I²C [7]



Rysunek 4.7 Wartość rezystorów podciągających uzależniona od wartości pojemności magistrali I²C [3]

Nazwa	Kod	Opis
SEARCH ROM	0xF0	Inicjowanie procesu wyszukiwania urządzeń w celu odczytania identyfikatora wszystkich urządzeń podpiętych do magistrali.
READ ROM	0x33	Możliwość odczytania identyfikatora urządzenia w przypadku, kiedy do magistrali podłączone jest jedno urządzenie.
MATCH ROM	0x55	Możliwość adresowania dowolnie wybranego urządzenia na magistrali. Po MATCH ROM zostaje wysłany 8-bajtowy identyfikator urządzenia.
SKIP ROM	0xCC	Umożliwia wybranie wszystkich dostępnych urządzeń podłączonych do sieci, bez względu na identyfikator. Dzięki temu, można zlecić czujnikom odczyt temperatury.
ALARM SEARCH	0xEC	Analogiczne działanie, jak w przypadku SEARCH ROM, jednak reagują jedynie urządzenia z ustawioną flagą alarmu.

Tabela 4.1 Lista poleceń 1-Wire obsługiwanych przez wszystkie urządzenia slave.

Rozdział 5

Pomiary zużycia energii elektrycznej

Ważnym aspektem, który chciano sprawdzić po zrealizowaniu części technicznej niniejszej pracy inżynierskiej, było wykonanie pomiarów zużycia energii elektrycznej. Zbadano zbiornik akwarystyczny niewyposażony w sterownik oraz zbiornik wyposażony w urządzenie sterujące. Jako założenie przyjęto, że zbiornik wyposażony jest w trzy elementy. Zalicza się do tego oświetlenie, filtr oraz grzałkę. Do testów wykorzystano zbiornik o pojemności 40 L. Jest to zbiornik, w którym znajdowała się jedynie woda, bez roślin i ryb. Zdecydowano się na taki zbiornik, aby zweryfikować poprawność działania wszystkich urządzeń.

Do pomiarów wykorzystano miernik energii zużycia energii elektrycznej wpinany bezpośrednio do gniazda elektrycznego. Na rysunku 5.1 można zobaczyć urządzenie, które zostało wykorzystane podczas testów. Miernik jest w stanie monitorować aktualne zużycie energii w zakresie od 15 do 3000 W, rejestruje najwyższy oraz najniższy pobór mocy. Jednak najważniejszą funkcją jest możliwość rejestrowania poboru energii w pewnym przedziale czasowym, który w przypadku pomiarów niniejszego systemu wynosił 24 godziny. Należy wspomnieć również, że dokładność pomiaru do 2000 W wynosi ± 2 W, ± 2 %. Na rysunkach 5.3 oraz 5.2 można zobaczyć wykorzystane urządzenia akwarystyczne. Grzałka, która została wykorzystana w testach nie posiadała wbudowanego termostatu. Wyłączenie było możliwe jedynie w przypadku odłączenia grzałki od źródła zasilania. Zastosowany filtr był małą jednostką, o wydajności wynoszącej 260 L/h. Filtracja odbywała się jedynie w sposób mechaniczny. Jako oświetlenie wykorzystano dwie diody typu *power led*. Deklarowana moc na tablicy znamionowej grzałki miała wartość 15 W, dla filtra jest to odpowiednio 4,2 W, a moc pojedynczej diody wynosiła 10 W.

Przygotowania do badania rozpoczęto od napełnienia zbiornika wodą. Łączna objętość wody wyniosła 40 L. Akwarium umieszczono na podłożu o dobrych właściwościach term izolacyjnych. Następna czynność polegała na umieszczeniu urządzeń akwarystycznych w odpowiednich miejscach. Filtr oraz grzałka zostały umieszczone blisko siebie. Celem tego zabiegu było rozprowadzanie podgrzanej wody przez filtr po całym zbiorniku. Miało to na celu osiągnięcie równomiernej temperatury wody w akwarium. W następnej kolejności zmierzono temperaturę w pomieszczeniu za pomocą czujnika DS18B20. Uzyskano wartość to $20,65^{\circ}\text{C}$, z kolei temperatura wody, pół godziny po napełnieniu wyniosła $24,65^{\circ}\text{C}$. Podczas badania przyjęto następujące założenia. Cykl dzień i noc wynosił po 12 h. Jest to zbliżone do cyklu występującego w naturalnym środowisku ryb. W trakcie doby założono dwie pory karmienia, w godzinach porannych oraz popołudniowych. Grzałka z powodu braku termostatu działała przez całą dobę. Pomiar rozpoczęto w południe i trwał całą dobę, aby wykonać pełny cykl. W trakcie badania zasymulowano dwa karmienia, podczas których wyłączono filtr. Łączna suma przerw w pracy filtra wyniosła w przybliżeniu 1,5 h.

Diody z planowanych 12 h pracowały przez 13,5 h. Jest to częsta sytuacja w przypadku sterowania ręcznego, aby utrzymać dokładne ramy czasowe. Jest to sytuacja, która jest niewskazana w przypadku roślin i zwierząt, ponieważ rozregulowuje to zegar biologiczny. W przypadku żywienia, występują naturalne okresy głodu, które mogą wynosić nawet dwa tygodnie. W przypadku akwarium, zaleca się, aby robić jeden dzień bez karmienia w ciągu 1-2 tygodni, co pozytywnie wpływa na ryby. Głodówki nie zaleca się w przypadku narybku, ponieważ młode ryby potrzebują znacznych ilości substancji odżywczych do prawidłowego rozwoju. Po upływie 24 h zatrzymano pomiar i odczytano zmierzoną wartość. Miernik zużycia energii wskazał 0,78 kWh. Należy pamiętać, że pomiar jest obarczony błędem $\pm 2\%$. Sprawdzono także temperaturę wody, która zmalała o $3,28\text{ }^{\circ}\text{C}$ do $21,82\text{ }^{\circ}\text{C}$. Temperatura panująca w pomieszczeniu utrzymała się na zbliżonym poziomie, co mieści się w dokładności pomiarowej czujnika pomiarowego.

Po przeprowadzeniu badania akwarium wyposażonego w podstawowe urządzenia, przyszła kolej na dodanie sterownika wraz z czujnikami. Na początku zmierzono temperaturę w pomieszczeniu, która wyniosła $19,71\text{ }^{\circ}\text{C}$. Temperatura wody wyniosła $20,43\text{ }^{\circ}\text{C}$. W następnej kolejności zdefiniowano parametry, jakie mają panować w zbiorniku. Do tego celu wykorzystano graficzny interfejs użytkownika. Oczekiwaną temperaturę wody ustawiono na poziomie $21,50\text{ }^{\circ}\text{C}$. Dla oświetlenia obliczono, że natężenie oświetlenia powinno wynieść 2534 lx. Dało to w przeliczeniu 40 lumenów na litr wody, co jest wartością zalecaną w przypadku najbardziej wymagających gatunków roślin. Diody emitowały światło przez 12 godzin, od 7:30 do 19:30. Cykl włączenia oraz wyłączenia został wykonany prawidłowo. W trakcie testu filtr został wyłączony dwukrotnie, aby zasymulować pory karmienia. Po upływie doby sprawdzono ilość pobranej energii, która wyniosła 0,628 kWh. Głównymi elementami, które pozytywnie wpłynęły na bilans energetyczny jest oświetlenie oraz grzałka. Elementem, który zwiększał zużycie energii w takiej konfiguracji jest sterownik, który pobierał 7,4 W. W skali roku daje to zużycie na poziomie 63,948 kWh. Przyjmując średnią cenę za kilowatogodzinę, która w momencie badania wyniosła 0,77 zł, to koszt pracy niniejszego sterownika w skali roku wyniesie około 50 zł.

Zastosowanie sterownika przyniosło pozytywny skutek, ponieważ zużycie energii zmniejszyło się o 0,152 kWh. W ciągu roku daje to szacunkowe oszczędności na poziomie 42,72 zł, zakładając bezawaryjną i ciągłą pracę. Im akwarium będzie większe tym bilans energetyczny będzie lepszy.



Rysunek 5.1 Miernik wykorzystany podczas testów



Rysunek 5.2 Grzałka wykorzystana podczas testów



Rysunek 5.3 Filtr wewnętrzny wykorzystany podczas testów



Rysunek 5.4 Akwarium, do którego docelowo zostanie użyty sterownik

Rozdział 6

Podsumowanie

Niniejsza praca miała na celu stworzenie systemu sterowania, który znajdzie zastosowanie w obszarze, jakim jest akwarystyka. Zbiornik akwarystyczny dzięki stworzonemu systemowi można określić mianem obiektu automatyzacji. Projekt obejmował szeroki zakres wiedzy. Podczas budowy urządzenia niezbędna okazała się wiedza z zakresu: akwarystyki, elektroniki oraz programowania w językach C i C++. Pozwoliło to na stworzenie kompletnego urządzenia od podstaw i realizację zadań wedle określonych potrzeb. Uzyskany efekt był zadowalający, dlatego sterownik zostanie wykorzystany do zarządzania akwariem, w którym znajdują się żywe organizmy. Docelowy zbiornik akwarystyczny jest widoczny na rysunku 5.4. Jest to akwarium roślinne, w którym występują gatunki z biotopów Ameryki Łacińskiej oraz Azji. Należą do nich: skalary (z łac. *Pterophyllum scalre*), bocje wspaniałe (z łac. *Chromobotia macracanthus*), zbrojniki niebieskie (z łac. *Ancistrus dolichopterus*) oraz akary błękitne (z łac. *Andinoacara pulcher*). W celu przystosowania sterownika do wcześniej wymienionego akwariem należy wymienić diody. Wybrany model musi cechować lepsze oddawanie barw oraz większa moc, aby dostarczyć zalecaną liczbę lumenów na litr wody. Należy podkreślić, że sterownik poprawia bilans energetyczny. Im akwarium posiada większą pojemność, tym oszczędności będą większe. Zostało to potwierdzone przez badania, które zostały przeprowadzone w ramach pracy inżynierskiej. W celu zwiększenia stopnia automatyzacji można wprowadzić automatyczne karmienie, co zlikwiduje konieczność ingerencji człowieka. W związku z wprowadzeniem takiego usprawnienia, cykl w akwariem stanie się ściśle określony, co pozytywnie przełoży się na zdrowie ryb. Takie rozwiązanie pozwala na dłuższą nieobecność właściciela, co jest dużą zaletą. Przydatnym rozwinięciem, jakie można dodać, polega na przeniesieniu całego systemu na serwer. Idzie za tym szereg korzyści. Możliwy będzie wtedy podgląd parametrów akwariem z dowolnego miejsca na ziemi, pod warunkiem dostępu do internetu. W takiej sytuacji warte uwagi stanie się dodanie kamery, która pozwoli na podgląd akwariem i wizualną ocenę stanu zbiornika. Kolejną ścieżką rozwoju jaką warto rozważyć jest dodanie inwertera. Urządzenie tego typu jest w stanie wygenerować z napięcia dostarczanego przez akumulator, napięcie przemiennie o wartości 230 V. Zapewni to energię dla najważniejszego urządzenia w akwariem, jakim jest filtr, w przypadku problemów z siecią elektroenergetyczną. Zasilanie awaryjne można rozważyć dla oświetlenia ze względu na zastosowaną technologię LED. W przypadku ogrzewania, takie rozwiązanie nie ma zbyt dużego sensu, ponieważ grzałki posiadają stosunkowo dużą moc. Osiąga ona nierzadko wartość 200 W lub więcej. Spowoduje to zbyt szybkie rozładowanie akumulatorów, nawet w przypadku wykorzystania kilku egzemplarzy. Następnym rozwinięciem systemu, które można zrealizować, polega na rozszerzeniu układu o kolejny sensor. Do jego zadania będzie należał pomiar pH wody.

Podczas tworzenia urządzenia pojawiały się problemy. W przypadku mikrokomputera Raspberry Pi pojawił się problem podczas konfiguracji środowiska Qt, wyniknęło to ze zmiany komend instalacyjnych i na tamtą chwilę nie istniały poradniki dotyczące nowego sposobu instalacji. Podczas tworzenia części związanej z elektroniką nie napotkano żadnych problemów. Część związana z mikrokontrolerem wymagała najwięcej uwagi i czasu, ze względu na stosunkowo dużą liczbę funkcji. Projekt mógł zostać wykonany jedynie w oparciu o mikrokomputer, jednak ze względu ilość dostępnego czasu, konieczne okazało się wykorzystanie mikrokontrolera z rodziny STM. Cały system oraz programy są konstruowane w taki sposób, aby poprawki oraz modyfikacje były wprowadzane z łatwością.

Załącznik A

Do pracy załączono płytę DVD zawierającą w poszczególnych katalogach:

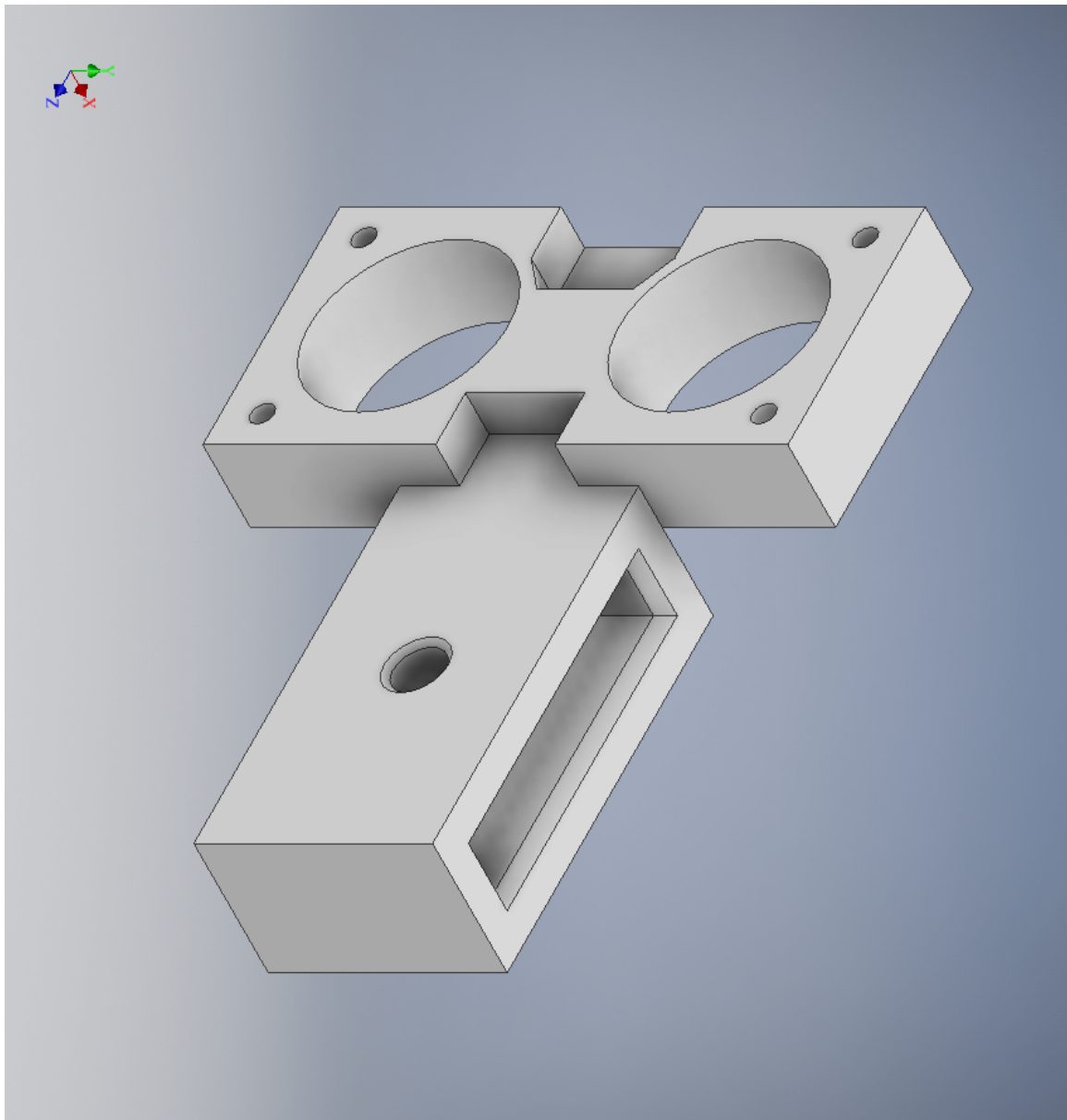
`/Praca_inzynierska.pdf` — wersja cyfrowa pracy,

`/Kod_zrodlowy_aplikacji_graficznejj` — kod źródłowy programu działającego na mikrokomputerze Raspberry Pi

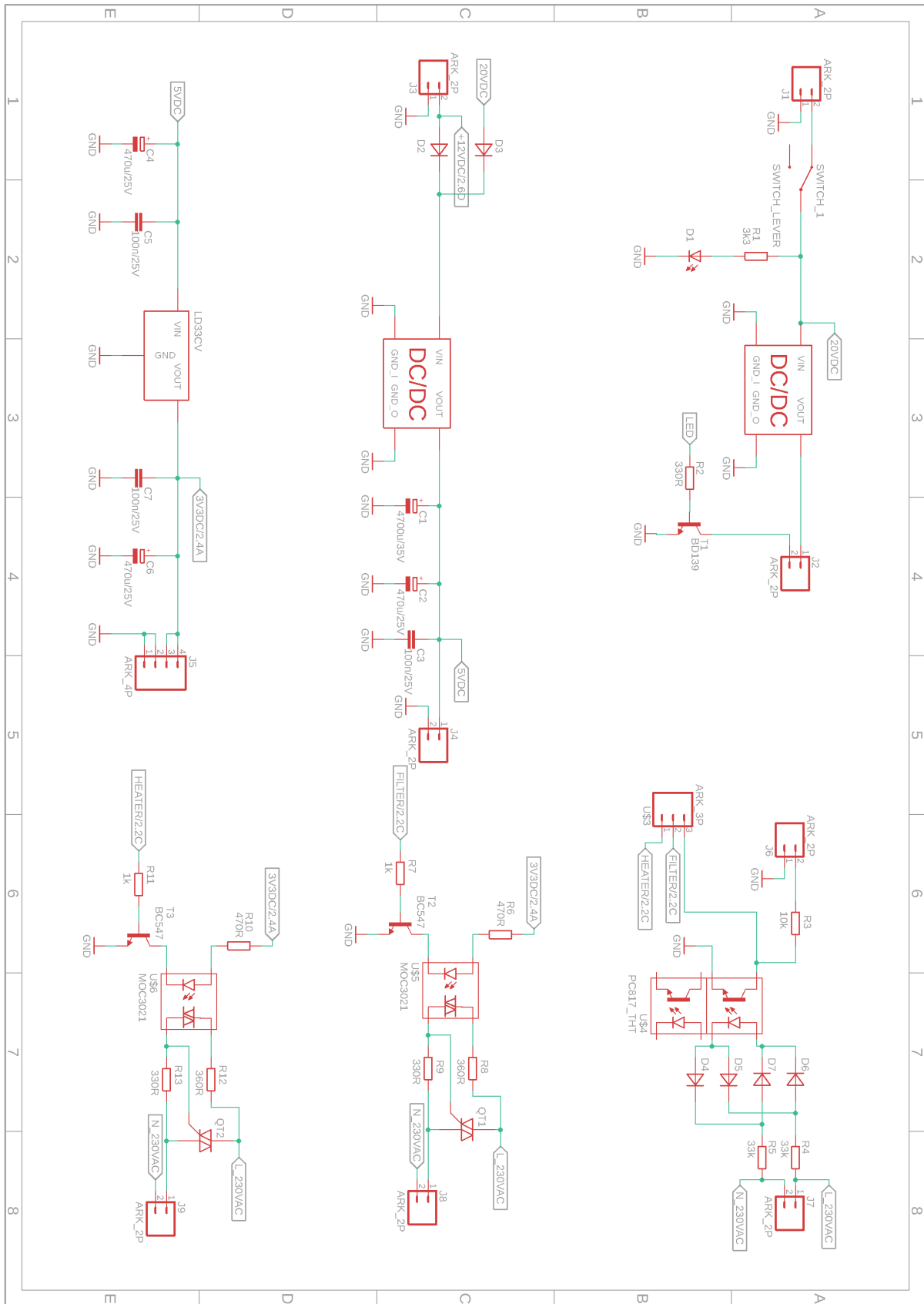
`/Kod_zrodlowy_mikrokontrolera` — kod źródłowy oprogramowania dla mikrokontrolera STM32

`/Schematy_elektroniczne` — schematy obwodów elektronicznych

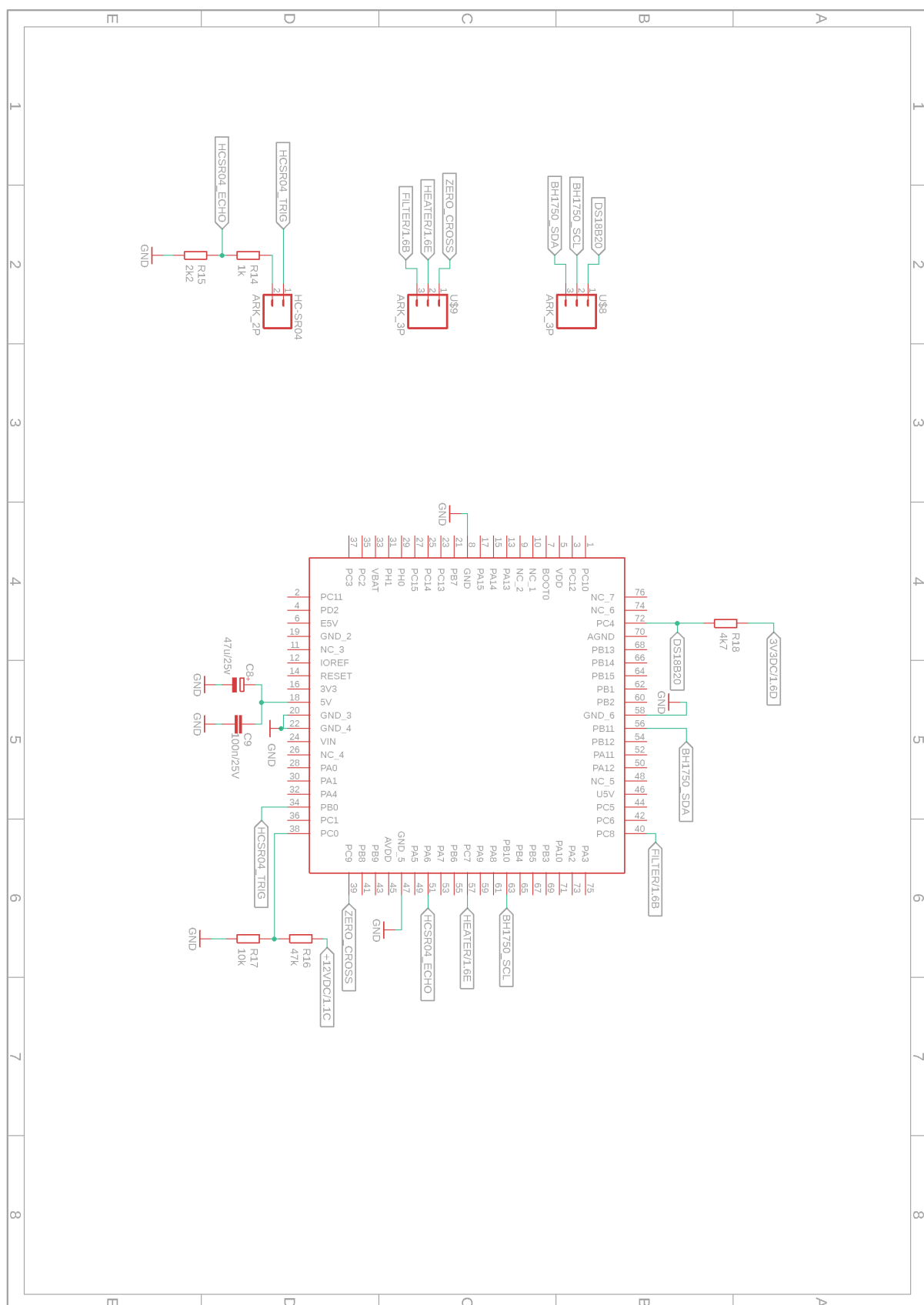
Załącznik B



Rysunek 1 Uchwyt na czujnik ultradźwiękowy wykonany w programie Autodesk Inventor 2023.



Rysunek 2 Schemat elektroniczny 1.



Rysunek 3 Schemat elektroniczny 2.

Literatura

- [1] *Raspberry Pi 4 Model B*. <https://datasheets.raspberrypi.com/rpi4/raspberry-pi-4-datasheet.pdf>.
- [2] AQUAEL sp. z o.o. Historia akwarystyki. Skąd przyplłynęła do nas pierwsza rybka? <https://www.aquael.pl/blog/akwaria/historia-akwarystyki/>, 2021.
- [3] R. Arora. *I²C bus pullup resistor calculation*. Texas Instruments, 2015. https://e2e.ti.com/cfs-file/__key/communityserver-discussions-components-files/6/I2C-Bus-Pullup-Resistor-Calculation.pdf.
- [4] Botland. Czujnik natężenia światła BH1750. <https://botland.com.pl/czujniki-swiatla-i-koloru/2024-czujnik-natezenia-swiatla-bh1750-5904422373283.html>, 2022.
- [5] Botland. Ultradźwiękowy czujnik odległości HC-SR04 2-200cm. <https://botland.com.pl/ultradzwiekowe-czujniki-odleglosci/1420-ultradzwiekowy-czujnik-odleglosci-hc-sr04-2-200cm-5903351241366.html>, 2022.
- [6] domowe-akwarium.pl. Parametry wody w akwarium - pH, Azotany, Fosfor, itp. <https://domowe-akwarium.pl/akwarium/chemia-w-akwarystyce>, 2022.
- [7] T. Francuz. Język c dla mikrokontrolerów avr. strony 531–535, 569–573. Helion, 2015.
- [8] N. Kularatna, J. Fernando. A supercapacitor technique for efficiency improvement in linear regulators. *Conference of IEEE Industrial Electronics*, strony 132–135, San Jose, CA, USA, 2009.
- [9] S. Prata. *C primer plus*. Pearson Education Ltd, 2014.
- [10] Sklep EVW Elektronika. DS18B20 czujnik temperatury wodoodporny. <https://evw-elektronika.pl/sklep/elektronika/czujniki/ds18b20-czujnik-temperatury-wodoodporny/>, 2022.
- [11] STMicroelectronics. *LF33CV 3,3V, 0.5A Very low drop voltage regulator with inhibit function*, 2017. <https://www.st.com/resource/en/datasheet/cd00000546.pdf>.
- [12] Trzmiel Czerwiński Spółka Jawna. Jakie oświetlenie do akwarium? <https://trzmielradzi.trzmiel.com.pl/jakie-oswietlenie-do-akwarium/>, 2021.