

Architektura systemu Host-Panel – Grupa 7

PROTOKÓŁ UART:

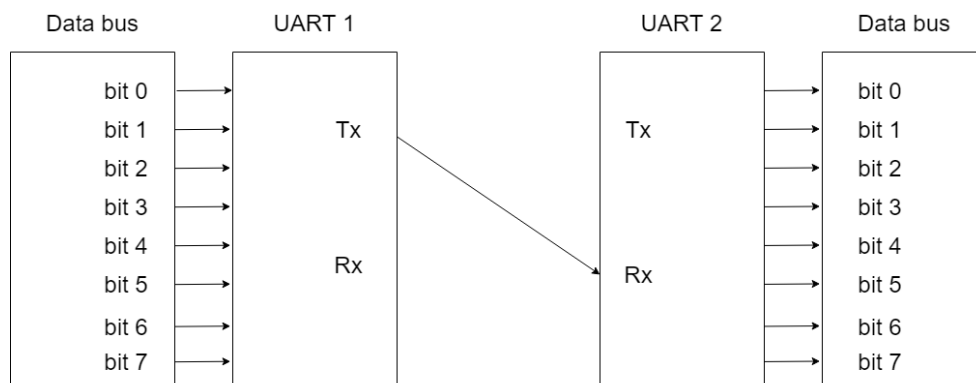
Komunikacja zestawu ewaluacyjnego z komputerem została zrealizowana za pomocą konwertera USB-UART FT232RL (wirtualny port COM).

UART oznacza uniwersalny odbiornik/transmitter asynchroniczny. Jest to fizyczny obwód w mikrokontrolerze lub samodzielny układ scalony. Głównym celem UART-u jest nadawanie i odbieranie danych szeregowo.

W komunikacji UART-u, dwa UART-y komunikują się bezpośrednio ze sobą. Do transmisji danych niezbędne są tylko dwa przewody. Dane są przesyłane od pinu TX nadawczego UART-u do pinu RX odbierającego UART-u. UART-y przesyłają dane asynchronicznie, co oznacza, że nie ma sygnału zegara do synchronizacji transmisji bitów. Zamiast sygnału zegarowego, transmitujący UART dodaje bity startu i stopu do przesyłanego pakietu danych. Definiują one początek i koniec pakietu danych, dzięki czemu „odbierający” UART wie kiedy rozpocząć odczyt. Gdy odbierający UART wykryje bit startowy, zaczyna odczytywać przychodzące bity z określoną częstotliwością znaną jako szybkość transmisji. Szybkość transmisji danych jest miarą szybkości transmisji danych, wyrażoną w bitach na sekundę (bps). Oba UART-y muszą pracować z mniej więcej tą samą szybkością transmisji.

Pierwszym krokiem w transmisji jest odebranie danych z szyny danych(data bus). Są one przekazywane do „nadawczego” UART-u w formie równoległej. Następnie do uzyskanych bitów dodawane są następujące bity: bit startowy, opcjonalny bit parzystości i bit stopu. Ramka danych może zawierać od 5 do 8 bitów, jeśli zastosowany jest bit parzystości(przy jego braku 9). Bit parzystości jest sposobem na to, aby odbierający UART mógł stwierdzić, czy jakiegokolwiek dane uległy zmianie podczas transmisji.

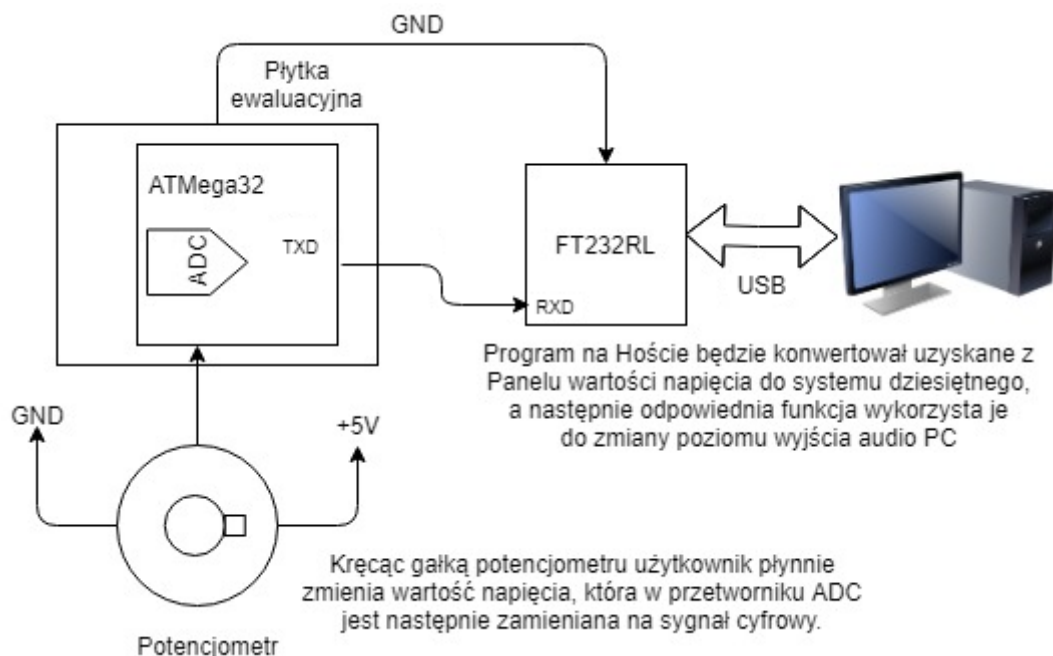
Następnie pakiet danych jest wysyłany seryjnie, bit po bicie na pinie Tx. Odbierający UART odczytuje pakiet danych bit po bicie na swoim pinie Rx. Potem dane są ponownie konwertowane do postaci równoległej, a bity specjalne(startu, stopu, parzystości) zostają usunięte. Ostatecznie „odbierający” UART przekazuje pakiet danych równoległe do podłączonej szyny danych.



Na powyższym rysunku jest przedstawiona transmisja danych w jedną stronę. W naszym przypadku transmisja może zachodzić w obie strony (od Hosta do Panelu i odwrotnie). Linie TXD i RXD są na naszej płytce ewaluacyjnej skrzyżowane ze sobą.

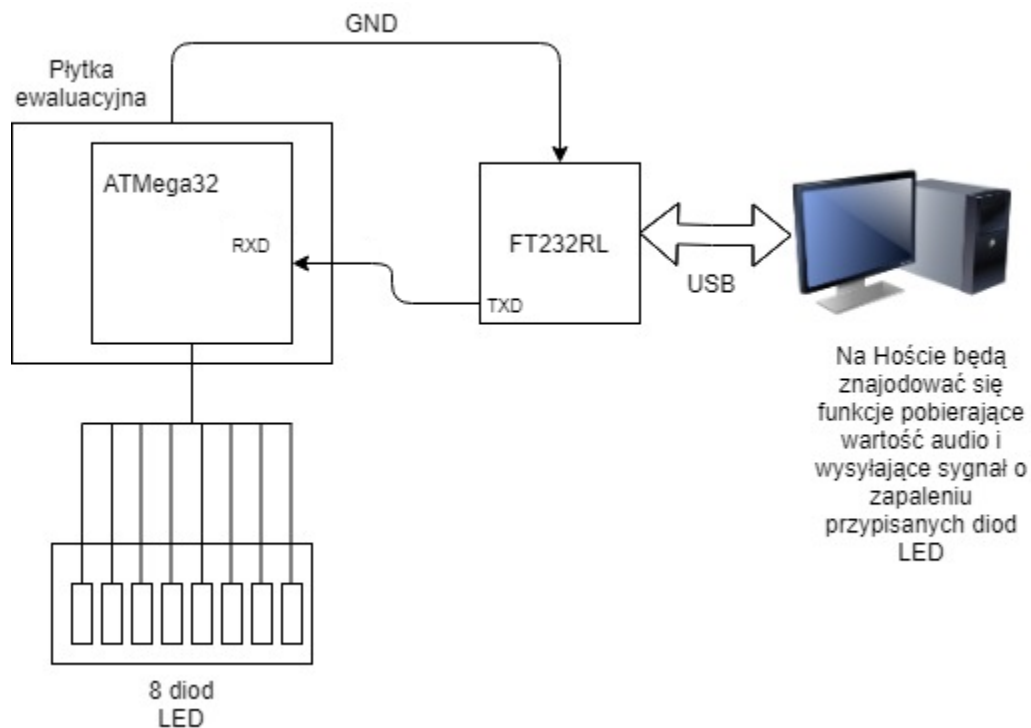
1. Możliwość kontrolowania poziomu wyjścia audio hosta za pomocą potencjometru na panelu (wartość z ADC):

Do wykonania tej funkcjonalności użyjemy potencjometru. Potencjometr to tak naprawdę regulowany rezystor z tą różnicą, że wartość napięcia jest zmieniana płynnie, a nie w sposób stały. Suwak potencjometru dzieli jego ścieżkę oporową na dwa połączone szeregowo rezystory, których suma rezystancji pozostaje stała. Taki potencjometr jest podłączany w roli dzielnika napięcia do przetwornika analogowo-cyfrowego. Dwa pozostałe wyjścia potencjometru będą podłączone do zasilania – odpowiednio jedno do VCC, a drugie do GND. Przetwornik analogowo-cyfrowy umożliwi nam zamianę sygnału analogowego na sygnał cyfrowy. Podsumowując całość sygnału analogowego będzie regulowana przez kręcenie gałką potencjometru przez użytkownika, a następnie sygnał ten będzie konwertowany na sygnał cyfrowy. z ADC zostanie następnie przesłana na Hosta poprzez UART. Na Hostcie odpowiednia funkcja dokona konwersji uzyskanej wartości na liczbę w zakresie od 0 do 100. Następnie kolejna funkcja wykorzysta uzyskaną wartość do zmiany wartości systemu audio. Funkcje będą działały w pętli i najprawdopodobniej z pewnym opóźnieniem (delay, itd.). Oprócz tego funkcje będą ignorowały bardzo niewielkie zmiany wartości napięcia, aby dokonać regulacji audio hosta tylko w przypadku, gdy użytkownik faktycznie zakręcił gałką potencjometru.



2. Wyświetlanie aktualnego poziomu wyjścia audio hosta na ośmiu diodach LED panelu:

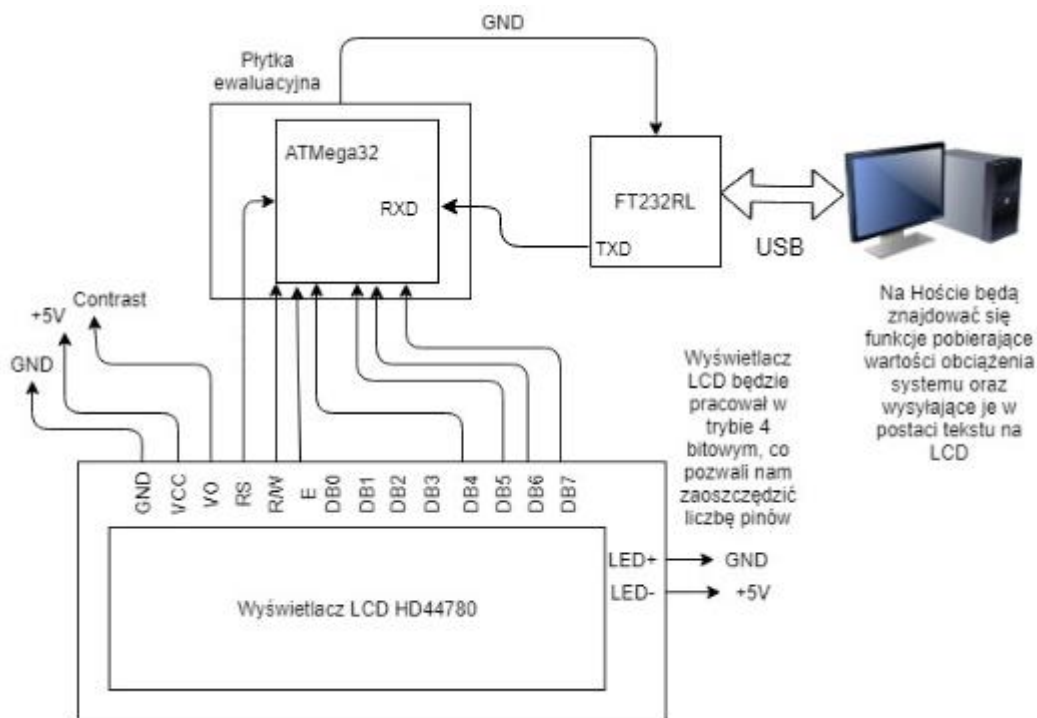
Wartość aktualnego poziomu audio hosta będzie pobierana przez funkcję w pętli w określonych odstępach czasowych. Ponieważ zakres głośności naszego Hosta wynosi od 0 do 100, a mamy tylko osiem diod LED na Panelu to każdemu poziomowi głośności zostanie przyporządkowany pewien zakres wynoszący 12.5(np. jedna dioda po osiągnięciu poziomu głośności 12,5, dwie diody po 25, itd.). Najprawdopodobniej będzie to zrealizowane za pomocą szeregu instrukcji warunkowych. Każda taka instrukcja w programie na Hoście będzie miała przypisane diody LED do wyświetlenia, a po spełnieniu określonego warunku sygnał z Hosta poprzez UART będzie przesyłany na płytkę ewaluacyjną i następnie spowoduje zaświecenie określonych diod LED.



3. Wyświetlanie obciążenia systemu hosta, użycia pamięci hosta i temperatury procesora hosta, na wyświetlaczu HD44780

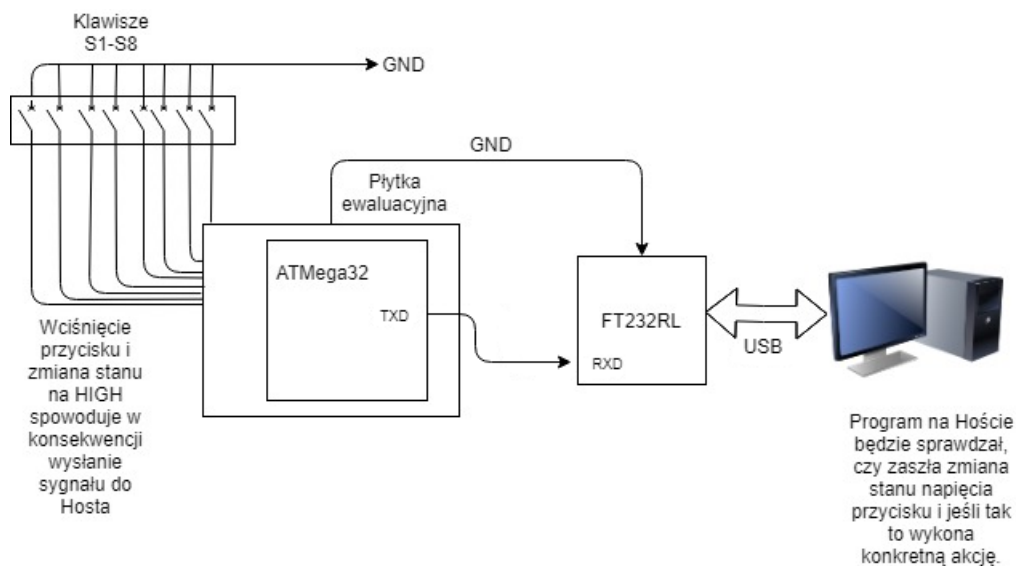
Wartości aktualnego obciążenia systemu hosta, użycia pamięci i temperatury będą pobierane za pomocą specjalnej funkcji w pętli w określonych odstępach czasowych w programie na Hoście. Wartości te następnie będą przesłane przez port UART do panelu. Ekran LCD w naszej płytce ewaluacyjnej jest wyposażony w 16-pinowe złącze i ma wymiary 2x16 znaków(2 wiersze po 16 znaków). Wyświetlanie na ekranie LCD będzie odbywać się w trybie 4-bitowym. Dzięki temu zaoszczędzimy liczbę pinów, ale jednocześnie transmisja danych potrwa dłużej niż w przypadku trybu 8-bitowego(wysyłane są dwa komunikaty, a nie jeden). Po włączeniu programu na naszym wyświetlaczu początkowo wyświetli się komunikat powitalny, a potem kolejno po sobie będą wyświetlane wartości obciążenia systemu hosta.

W miarę możliwości podczas budowy tej części systemu wykorzystamy obecne wraz z pakietem FreeRTOS biblioteki m.in. lib_hd44780.



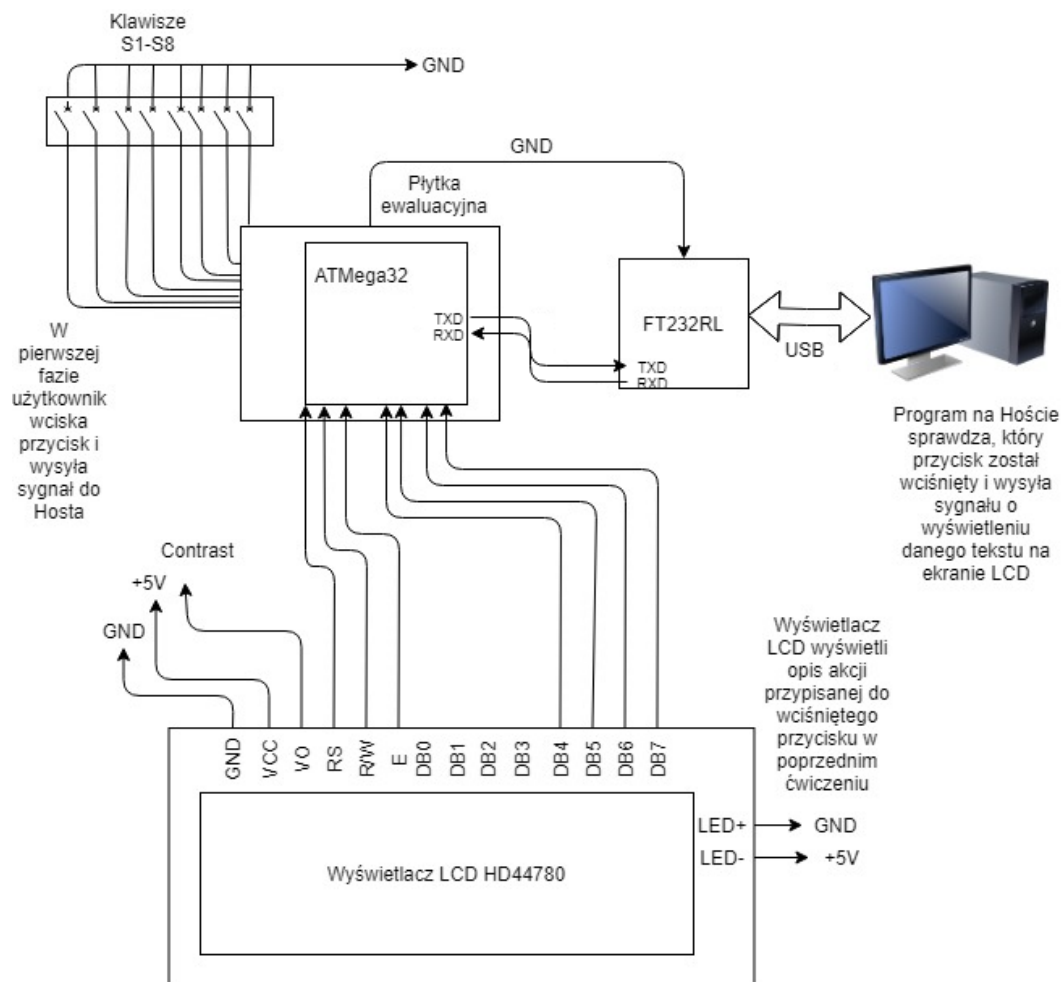
4. Klawisze skrótu S1... S8 uruchamiające dowolną akcję na hoście

Gdy każdy z przycisków od S1 do S8 jest otwarty(niewciśnięty) to nie ma połączenia między dwiema nogami przycisku i mamy stan HIGH. Przy zamknięciu(wciśnięciu) przycisku tworzy się połączenie między jego dwoma nóżkami i uzyskujemy stan LOW. Tak wygenerowane sygnały będą przesyłane za pomocą UART do naszego Hosta. Nasz program na Hoście będzie zawierał zmienną przypisaną do każdego przycisku od S1 do S8. Wartość każdej zmiennej ulegnie zmianie wtedy, gdy stan danego przycisku zmieni się z „off” na ‘on”(każde wciśnięcie). Bardzo możliwe, że będzie też zliczał ile razy miała miejsce zmiana stanu przycisku. Gdy wartość zmiennej przypisanej do przycisku ulegnie zmianie na Hoście zostanie wywołana konkretna akcja. Oprócz tego na Hoście użyjemy dodatkowej zmiennej, która będzie zapobiegała wielokrotnemu wykonaniu danej akcji przy pojedynczym wciśnięciu odpowiedniego przycisku.



5. Wyświetlanie opisów/pomocy klawiszy skrótów na wyświetlaczu HD44780

Ta część systemu będzie w zasadzie podzielona na dwie składowe. W pierwszej użytkownik wciska jeden z klawiszów od S1 do S8 umieszczonych na płytce ewaluacyjnej. W wyniku tworzy się połączenie między nóżkami przycisku i uzyskujemy stan HIGH. taki sygnał za pomocą UART zostanie przetransportowany z panelu do naszego Hosta. Program na Hoście będzie zawierał zmienne, z których każda będzie przypisana do poszczególnego przycisku od S1 do S8. Program będzie działał w pętli i odczytywał zmiany wartości stanu przycisku w określonych odstępach czasowych. Po zmianie wartości danej zmiennej komendy w poszczególnych instrukcjach warunkowych spowodują wyświetlenie przypisanego tekstu, który będzie opisywał, jaką akcję powoduje wciśnięcie danego przycisku(z poprzedniego podpunktu). Informacje te zostaną ponownie wysłane do panelu poprzez UART i wyświetlone na ekranie LCD w trybie 4 bitowym.



6. Wyświetlanie na diodzie RGB panelu bieżącego uśrednionego koloru strumienia video odtwarzanego na hoście. Konieczne jest uwzględnienie korekcji gamma. (tak jak funkcja Ambilight)

We wnętrzu diody RGB znajdują się trzy struktury świecące- czerwona, zielona oraz niebieska. Dioda RGB w naszym przypadku posiada cztery wyprowadzenia- najdłuższą nóżkę wyprowadzoną do masy, a pozostałe linie mogą pracować w trybie PWM.

Wysterowanie diody RGB odbędzie się poprzez zmianę współczynnika wypełnienia sygnału PWM, która przyjmuje wartości z zakresu od 0 do 255(dla zilustrowanie podając wartość 0 otrzymamy pełen przepływ prądu i największą intensywność świecenia danej barwy). Do pozyskania uśrednionego koloru strumienia wideo użyjemy dodatkowych bibliotek do przechwytywania screenshotów, który dla każdego piksela weźmie wartość koloru czerwonego, zielonego i niebieskiego, a następnie sumę ich wszystkich uśredni. Taka wyprowadzona wartość dla każdego koloru zostanie przekazana przez UART do naszego Hosta i razem z pozostałymi kolorami spowoduje wyświetlenie koloru, który będzie uśrednionym kolorem na monitorze(w miarę możliwości dla zilustrowania tej części systemu umieścimy płytkę ewaluacyjną za podłączonym do niej monitorem).

