

1 Cel projektu

Celem projektu jest zbudowanie klawiatury komputerowej w układzie ANSI TKL. Za pracę klawiatury odpowiadać będzie mikrokontroler, który zajmie się komunikacją z komputerem poprzez USB. Głównym wyzwaniem projektu jest stworzenie kompletnego układu mikrokontrolera oraz programu w postaci modułu, do którego można będzie przyłączyć matrycę przycisków.

2 Założenia projektu

Na podstawowe założenia projektu składają się:

- Układ klawiatury w układzie ANSI TKL
- Komunikacja z komputerem poprzez USB
- Pooling rate 500Hz lub więcej
- Programowy de-bouncing
- Wspomaganie adresowania matrycy przełączników przy pomocy układów 74AHC138
- Wsparcie NKRO

Na dodatkowe założenia projektowe składają się:

- Obsługa makr
- Sterowanie podświetleniem poprzez moduł
- Mini system operacyjny

3 Wybór rozwiązania projektowego

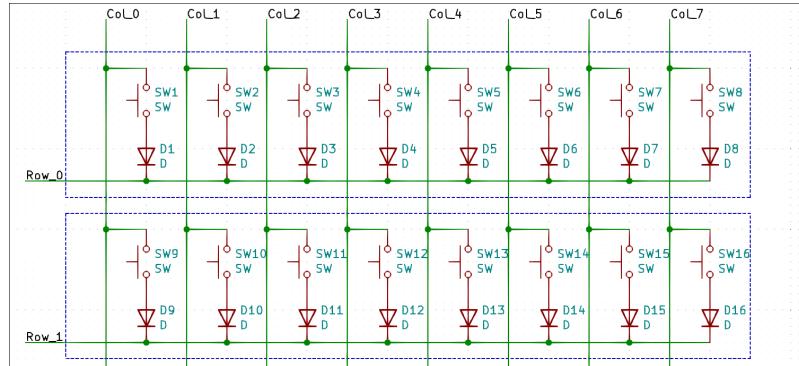
Praca klawiatury komputerowej w głównej mierze sprowadza się do ciągłego wykonywania następujących czynności przez mikrokontroler:

1. Pobierz stany przełączników (np. wykrycie stanu wysoki/niski na pinach)
2. Na podstawie danych, wygeneruj kolejkę zmian przycisków (downstroke i upstroke, scancode)
3. Prześlij dane do komputera/hosta

Oczywiście to jest najsuworsza pętla pracy klawiatury. Jako układy wykonawcze postanowiono wykorzystać układ ATMega32U4. Jest to mikrokontroler firmy Atmel (obecnie pod Microchip), który jest szeroko stosowany w układach hobbystycznych. Układ ma ten następujące cechy, decydujące o jego zastosowaniu:

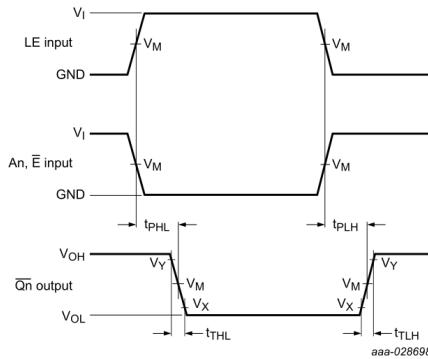
- Duża wydajność obliczeniowa (głównie instrukcje wykonywane w 1 cyklu zegara oraz maks 16MHz)
- Sprzętowe USB dające sporo możliwości
- Bogate wyposażenie tj. 4 liczniki, I²C, SPI, tryby uśpienia, 2 pełne 8-bitowe porty I/O
- Zintegrowany stabilizator 3,3V, wymagany dla komunikacji z USB
- Bogata dokumentacja i obecność gotowych rozwiązań OpenSource (GH60) opartych na tym układzie
- Przyjazna w lutowaniu obudowa TQFP44
- I przede wszystkim spora ilość projektów klawiatur OpenSource opartych na tym układzie

Wybrany mikrokontroler posiada 44 piny. Standardowy układ ANSI posiada 104 klawisze. Z góry widać, że bezpośrednie podłączenie przycisków do pinów mikrokontrolera nie wystarczy. Zamiast takiego podejścia stosuje się *matrycę przełączników*.



Rysunek 1: Przykład matrycy przełączników

Połączenie przełączników w matrycę pozwala na adresowanie wybranej kolumny przycisków, a następnie ich odczytu. Ponieważ podczas odczytu musi być aktywna tylko jedna kolumna, to atrakcyjnym jest zastosowanie dekoderów n do n^2 . Przykładowymi układami mogą być 74HC154 lub 74AHC138. Pierwszy układ to dekoder 4-do-16, zamienia 4 bitową wartość na wejściach na 1 z 16 na wyjściach. Aktywne wyjście przyjmuje stan niski, co pozwala na zastosowanie go bezpośrednio w układzie. Układ 74AHC138 jest tym samym układem co 154, ale 3-do-8. Oba układy posiadają możliwość łączenia w celu rozszerzenia ilości linii. Układ 154 byłby idealny, ponieważ 8×16 daje 128 klawiszy. Niestety, układ ten jest już stary i dostępny jedynie w starym procesie technologicznym HC(T), który posiada duże opóźnienia (mogą one przekroczyć 50ns) i pobiera więcej energii (chociaż i tak pobierana energia jest znikoma). 74AHC138 natomiast, jest dostępny w procesie AHC, który jest znacznie szybszy (wszelkie opóźnienia nie przekraczają 10ns) i łatwiej dostępny. Układ ten również można rozszerzyć do 5-do-32. W naszym przypadku jedynie potrzebne jest 16 linii, ale nic nie stoi na przeszkodzie, by układ rozszerzać.



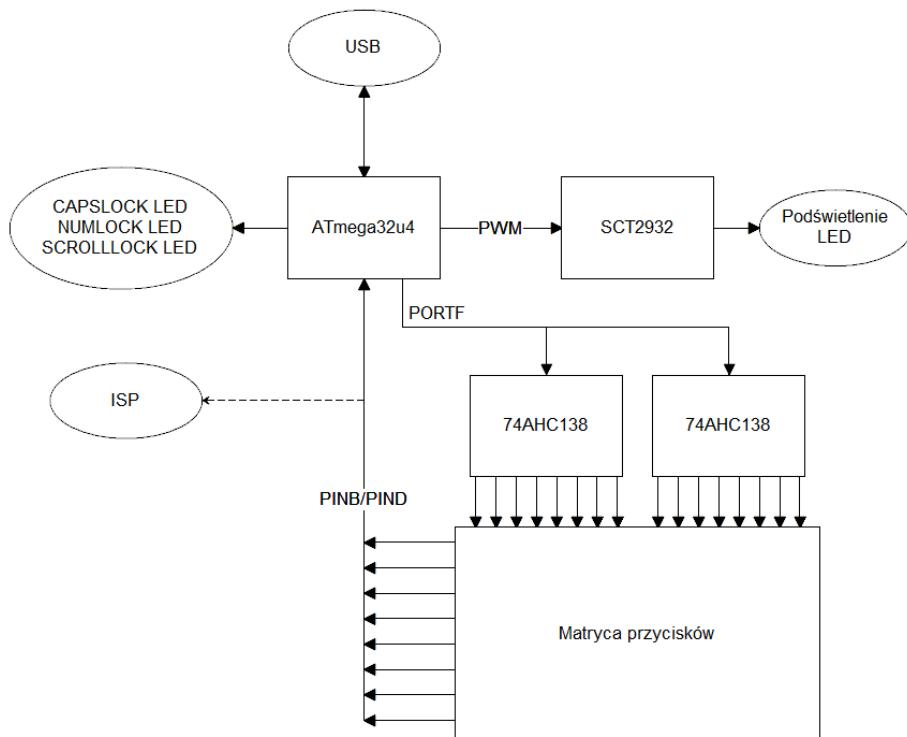
Rysunek 2: Diagram czasowy

Oznaczenie	Opis	Tpd	Tt	Warunki An do Qn
74HC4515	4-to-16 line decoder/demultiplexer with input latches;inverting	50ns	15ns	4,5V
74HC154	4-to-16 line decoder/demultiplexer	30ns	15ns	4,5V
74AHC138	3-to-8 line decoder/demultiplexer; inverting	10.1ns	N/A	4,5-5,5V

Producent	SKU	n-pin	Flash	SRAM	EEPROM	Wydajność	Napięcie	Peryferia
Atmel/Microchip	ATMega32U4	44	32KB	2,5KB	1KB	16MIPS	2,7-5,5V	USB JTAG SPI TWI USART ADC
NXP	MC9S08JM60	do 64	do 60KB	do 4KB	N/A	48MHz	2,7-5,5V	USB SPI TWI ADC RTC
Microchip	PIC18F47J13	44	128KB	3760B	N/A	12MIPS	2-3,6V	USB SPI TWI RTCC ADC

Co do mikrokontrolera to są jeszcze inne opcje. Poza mikrokontrolerami AVR8 firmy Atmel całkiem niemałą popularnością cieszą się układy innych firm, takie jak uC z serii PIC firmy Microchip oraz nieco bardziej skomplikowane (ale wydajniejsze) STM32 firmy ST. Z jakich powodów wybrany został uC ATMega32U4 to już zostało przedstawione.

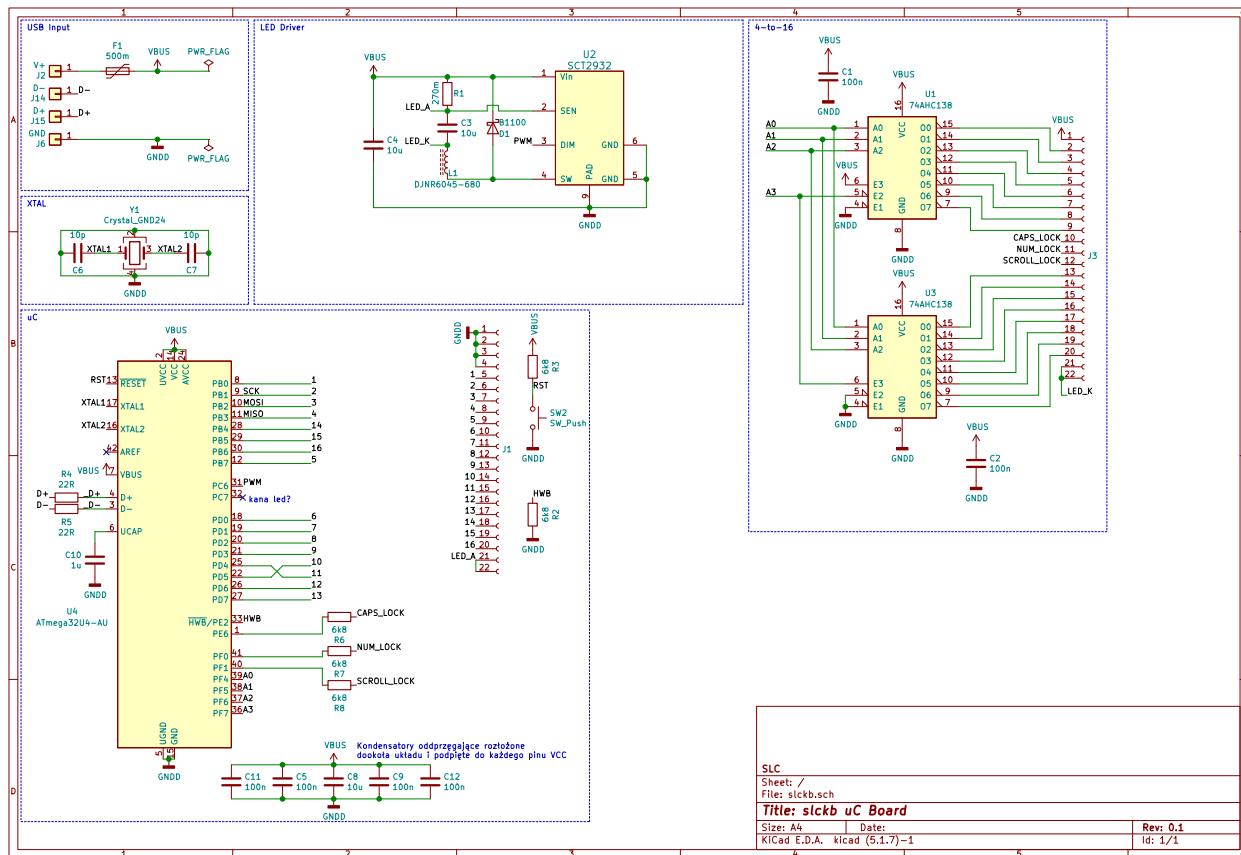
4 Schemat blokowy



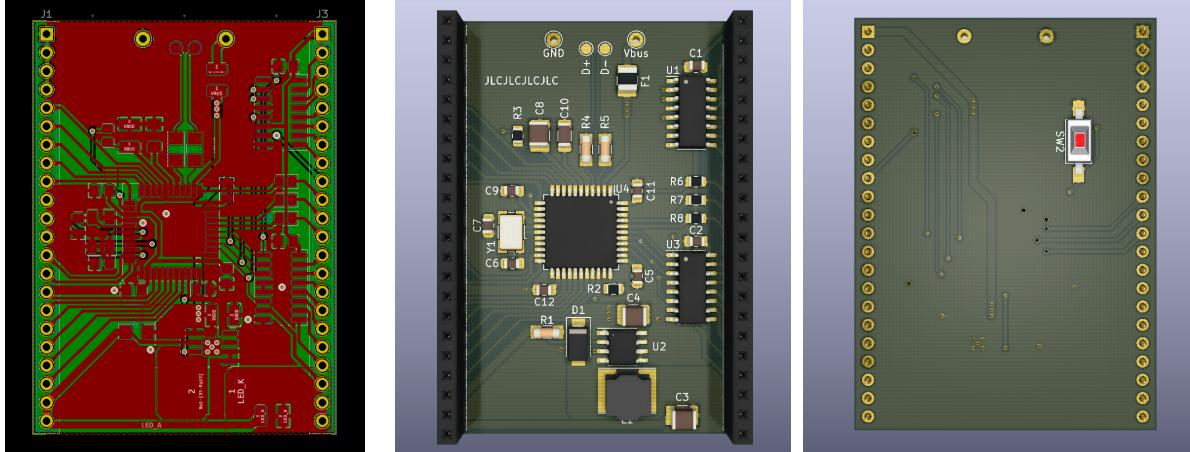
Schemat blokowy układu wygląda następująco:

- Sercem płytki jest mikrokontroler ATMega32U4
- 4 piny z portu F są wykorzystywane do adresowania matrycy przycisków
- Odczyt z macierzy odbywa się poprzez wykorzystanie jednego lub dwóch pełnych portów B/B+D
- Pin C6 jest wykorzystywany jako wyjście dla sygnału PWM i jest on podłączony bezpośrednio do driver'a podświetlenia
- Jeśli do odczytu macierzy wykorzystuje się tylko jeden port (B), to wówczas linie portu D mogą być wykorzystane do innych celów
- 3 piny uC są dedykowane do obsługi kontrolek Caps Lock, Num Lock, Scroll Lock

Powyzszy schemat blokowy zrealizowano w postaci schematu:



A następnie płytki drukowanej:



(a) Rysunek płytki

(b) Render płytki góra

(c) Render płytki dól

Tak jak widać, można zauważyc przycisk na spodzie płytki. Cała płytka ma pełnić rolę modułu, który jest wpinany w płytke PCB z przełącznikami. Sam rozstaw listw pinów nie jest przypadkowy, ponieważ wynosi on 1,5 cala, co jest równe 2u (u - unit - jednostka). Jeden u w terminologii klawiatur jest równe 0,75", i taka odległość dzieli centra przełączników. Innymi słowy moduł ten powinien zmieścić się pomiędzy rzędy przełączników. Z tego właśnie powodu jest przycisk na spodzie płytki, aby łatwo można było zresetować układ mikrokontrolera i przejść w tryb DFU (tryb w którym można programować układ bezpośrednio z USB). Dodatkowo na płytce znajduje się jeszcze bezpiecznik polimerowy 500mA.

5 Układ podświetlenia

Poza układami 74AHC138 na płytce znajduje się jeszcze układ SCT2932 firmy Starchip. Jest to jeden z wielu układów przeznaczonych do zasilania LED stałym prądem. Układ ten został wybrany z kilku następujących powodów:

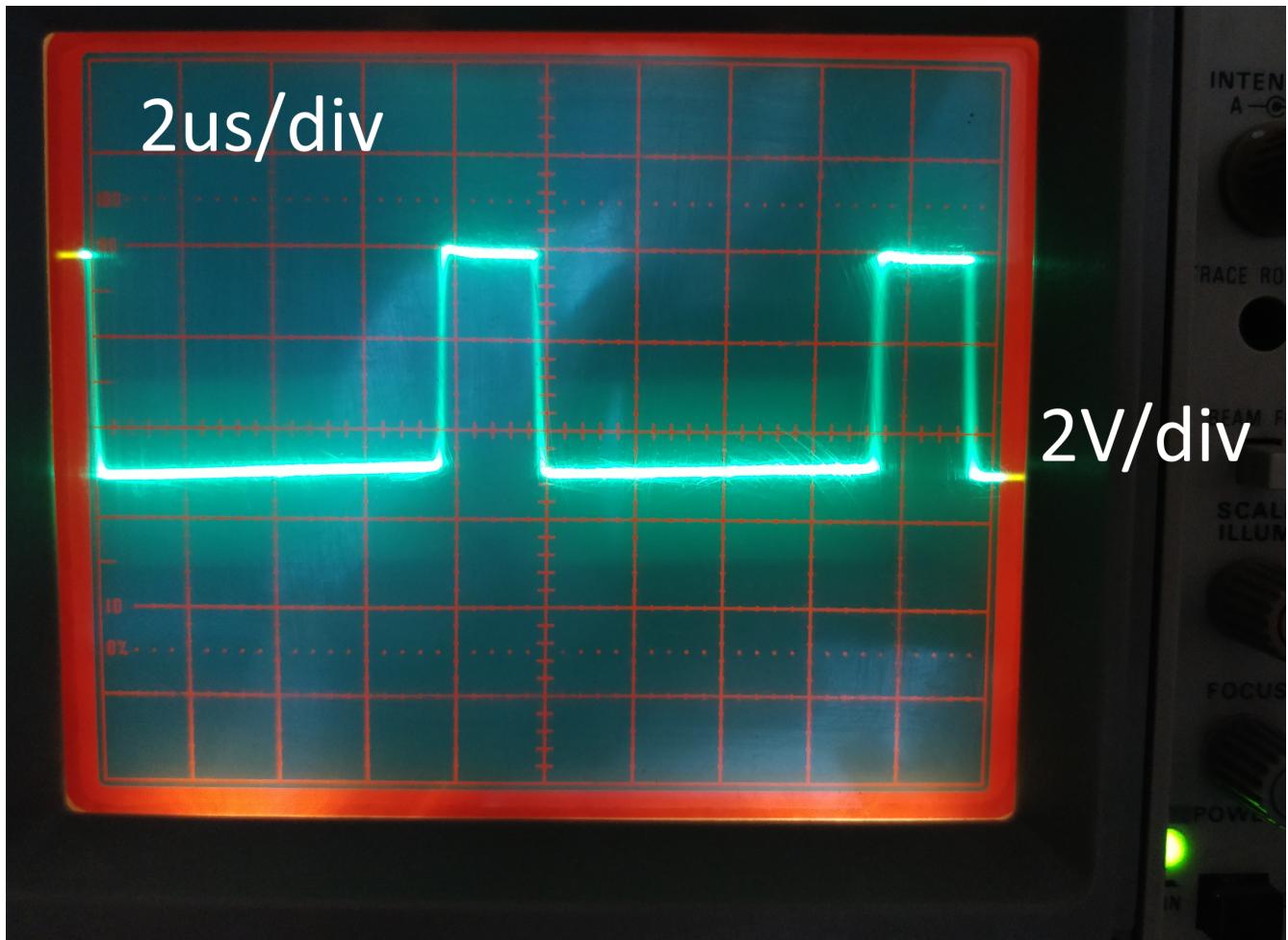
- Może pracować od 5 do 33V
- Bardzo prosta konstrukcja
- Możliwość sterowania jasnością poprzez PWM
- Zabezpieczenie termiczne, przeciwwzwarciowe i soft-start
- Wymaga tylko 4 komponentów zewnętrznych
- Możliwość pracy do 1,5A prądu wyjściowego
- Prąd jest ustalany poprzez dobranie rezystora
- 0,1V napięcia odniesienia - niski dropout

Układ domyślnie miał pracować z rezystorem 270mOhm co powinno dać 370mA prądu. Przy tej wartości został zbadany układ w celu ocenienia jego sprawności. Ostatecznie zaobserwowano:

- Sprawność bardzo mocno zależy od odsprzęgnięcia zasilania układu (C4), dlatego wykorzystano kondensator SMD MMLC 1210 X7R 10uF
- Sprawność zależy od różnicy między napięciem wejściowym zasilania a wyjściowym (im mniejsza różnica tym wyższa sprawność)

- Białe diody LED przy prądzie 370mA potrzebowaly 2,75V (przy niższym prądzie wartość ta będzie niższa oczywiście)
- Najwyższa zmieżona sprawność to 75%, jednakże należy mieć na uwadze, że oba kondensatory (C4 i C3) nie były dobrze połączone z układem

Teoretyczna sprawność powinna spokojnie zaczynać się od 80% dla wyższych prądów (powyżej 150mA). Wartości komponentów zewnętrznych są takie same jak te opisane w nocy katalogowej układu. Przy powyższych warunkach zmierzono również częstotliwość z jaką pracuje układ (zależy ona głównie od stosunku Vin do Vout) i wyniosła ona około 100kHz.



Rysunek 4: Napięcie na wyjściu tranzystora kluczującego. Stan niski oznacza przewodzenie

6 Program

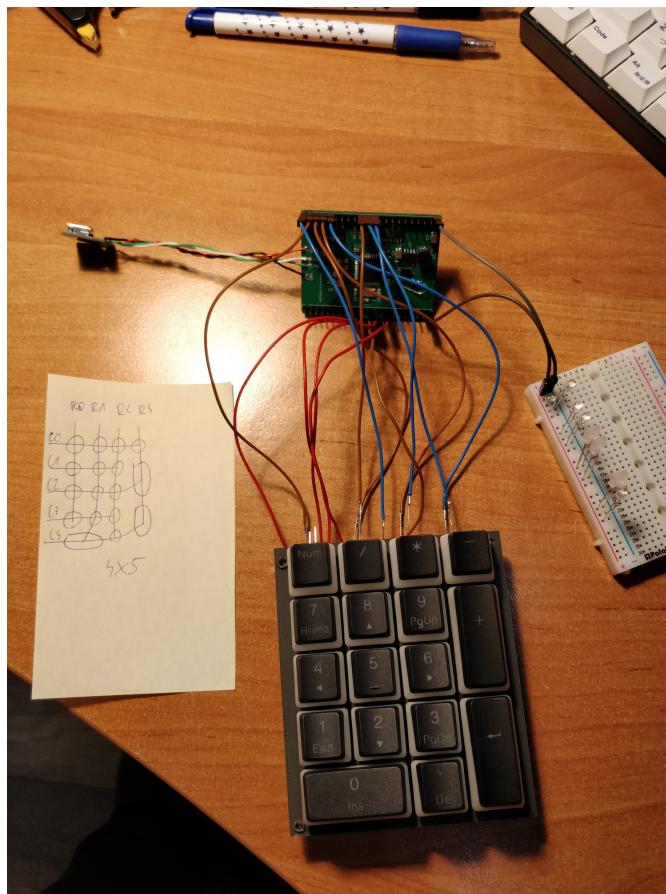
Napisanie programu dla klawiatury jest jednocześnie łatwym i trudnym zadaniem. Wszystko zależy od interfejsu klawiatura-komputer. Starsze klawiatury (i myszki) posługiwały się portem PS/2 (jeszcze starsze AT), który po mimo swoich zalet, został już prawie całkowicie wyparty. Główną zaletą PS/2 była prostota przesyłu danych po dwóch liniach. Warstwa fizyczna nie wymaga dedykowanego układu, a protokół jest relatywnie prosty do implementacji w programie.

Jeśli chodzi o USB to sprawa jest odwrotna. USB jest stworzone by być uniwersalne i przez to nawet najprostsza rzecz jak klawiatura i mysz wymagają pełnej implementacji protokołu, punktów końcowych, deskryptorów, ramek i pakietów. Z tego powodu postanowiono skorzystać z rozwiniętego projektu oprogramowania do klawiaturow opartych na układach AVR i SMT32: QMK. QMK to bardzo kompleksowe rozwiązanie do zaprogramowania układu klawiatury. Główną siłą QMK jest jego elastyczność. Każda implementacja układu mikrokontrolera jest rozróżniana w folderze `firmware/qmk_firmware/keyboards`. W tym folderze znajdują się pliki, które modyfikują domyślny program tak, aby pasował on do naszego rozwiązania. W naszym przypadku jest to `firmware/qmk_firmware/keyboards/slckb`. W domyślnej postaci użytkownik musi jedynie określić piny wykorzystywane do skanowania matrycy (plik `config.h`), komponenty, które mają zostać skompilowane i dołączone do programu (`rules.mk`), keymapa (tłumaczca fizyczne położenie przycisku w matrycy na nasze) i układ/przypisanie przycisków.

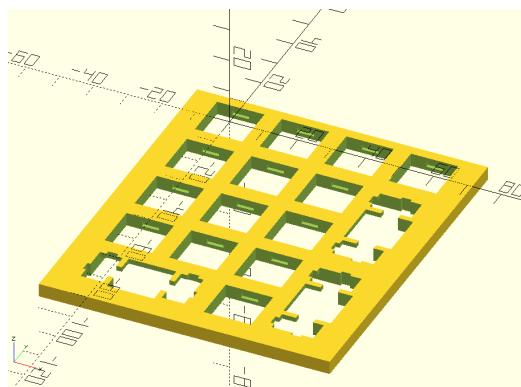
Aby zmodyfikować QMK do potrzeb projektu, trzeba zdefiniować własną wersję funkcji wykonujących dane czynności tj. inicjacja matrycy i jej skan. W tym celu wykorzystano dokumentację dostępną na stronie projektu: <https://qmk.fm/> Kod jest dobrze skomentowany i tłumaczy kolejne czynności. W pliku `config.h` można zdefiniować takie rzeczy jak: wymuszenie NKRO (false), interwał skanowania (1ms), czas debouncingu (3ms).

7 Prototyp

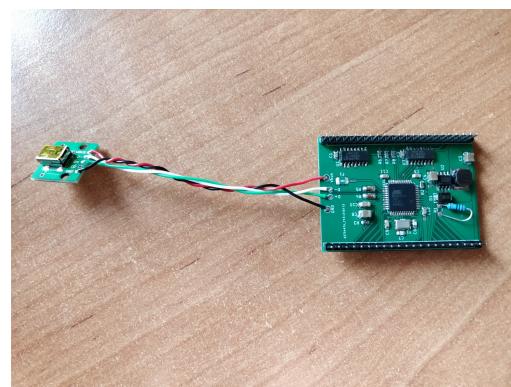
Po stworzeniu programu przyszedł czas na prototyp. Przy użyciu OpenSCAD stworzona została płytka testowa, która mieści 17 przełączników. Cała płytka ma kształt i układ numpada. Płytki ta została wydrukowana na drukarce 3D.



Rysunek 5: Złożony prototyp wraz ze schematem połączenia



(a) Model płytki montażowej

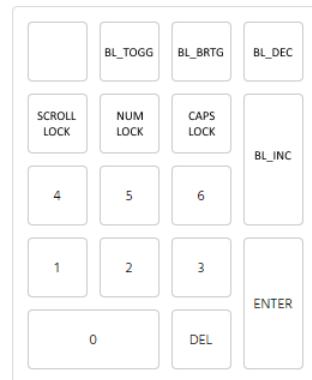


(b) Złożone i zlutowane PCB

W prototypie zdefiniowano 2 warstwy. Definicje warstw można znaleźć w pliku `firmware/qmk_firmware/keyboards/slckb/keymaps/default/keymap.c`



(a) Warstwa główna



(b) Warstwa FN

Diody od kontrolek ScrollLock, CapsLock oraz NumLock zostały umieszczone pod przyciskami / , * , - . Ponieważ cały prototyp jest zlutowany metodą p2p, dodanie diód podświetlenia wprowadziłoby duży balagan i możliwość zwarcia zasilania podświetlenia z obwodami matrycy. Z tego powodu za "podświetlenie" będzie służyć kilka diód połączonych równolegle w płytce stykowej. W ostateczności wszystko działa. Dzięki wykorzystaniu QMK zostały osiągnięte wszystkie założenia projektu za wyjątkiem układu klawiatury (numpad zamiast TKL). Jest to spowodowane delikatnym przerostem projektu oraz złego zarządzania czasem odnośnie prac nad poszczególnymi elementami projektu. Ostateczny produkt wraz z oprogramowaniem może zostać wykorzystane do budowy klawiatury o wymiarach matrycy 16x16, co daje 256 klawiszy. Oczywiście tak długo jak starczy pamięci programu oraz SRAM.